DRA
Hqtrs
NSG-7506

UNIVERSITY OF ILLINOIS
URBANA

AERONOMY REPORT
NO. 110

AN INVESTIGATION OF TURBULENT SCATTER FROM THE MESOSPHERE AS OBSERVED BY COHERENT-SCATTER RADAR

by
K. P. Gibbs
S. A. Bowhill

April 1, 1983

Library of Congress ISSN 0568-0581

Aeronomy Laboratory
Department of Electrical Engineering
University of Illinois
Urbana, Illinois

Supported by
National Aeronautics and Space Administration
National Science Foundation

UILU-ENG-83-2502

A E R O N O M Y   R E P O R T

N O. 110

AN INVESTIGATION OF TURBULENT SCATTER FROM THE MESOSPHERE AS OBSERVED BY COHERENT-SCATTER RADAR

by

K. P. Gibbs
S. A. Bowhill

April 1, 1983

Supported by
National Aeronautics
and Space Administration
Grant NSG 7506
National Science Foundation
Grants ATM 78-21765 and ATM 81-20371

Aeronomy Laboratory
Department of Electrical Engineering
University of Illinois
Urbana, Illinois

ABSTRACT

Turbulent scatter from the mesosphere is observed using the Urbana coherent-scatter radar. The variation in signal-to-noise ratio as a function of time-of-day is examined. The origin of scattering regions is investigated by comparing the variations in scattered power and Doppler velocity. Nighttime echoes are shown for periods of enhanced electron concentration. The spectrum of the returned signal is studied with a resolution of ten seconds. Spectral information is used to increase altitude resolution and observe the motion of scatterers.

The expected variation in signal-to-noise ratio with solar flux is observed. It is found that variations in the scattered power generally do not correspond to the gravity waves which are simultaneously observed. Turbulent layers are observed at altitudes with high shear in the horizontal velocity and at altitudes with low shear. The ten-second resolution is necessary to distinguish meteor echoes from echoes produced by the advection of a scattering layer through the radar beam. The width of the spectrum of the scattered signal increases with altitude. Fresnel scattering is indicated below 75 km and bulk scattering is indicated above. Scattered layers which move in pairs indicate the generation of turbulence by a Kelvin-Helmholtz instability. Acoustic wave motion is indicated in the data with ten-second resolution.

TABLE OF CONTENTS

LIST OF FIGURES

1. INTRODUCTION

1.1 General Introduction

The existence of over-the-horizon communication in the form of multiple-hop shortwave transmission gave rise to many of the earliest studies of the ionosphere. Propagation experiments at various frequencies were carried out, leading to the discovery of a scatter propagation mode both at tropospheric and mesospheric heights. With the advent of satellites, mesospheric scatter lost its importance as a communication technique. However, radio techniques have continued to play an important role in studies of the ionosphere. The development of one such technique, coherent-scatter radar, is detailed in Gibbs and Bowhill [1979]. In this study, the Urbana coherent-scatter radar is used to investigate scatter from the mesosphere. A more detailed description of this work is given below.

1.2 The Mesosphere

The earth's atmosphere is divided into regions based on the thermal properties, i.e., the temperature profile. From the earth s surface up to about 15 km, the temperature decreases with increasing altitude. This region, the troposphere, is therefore said to have a positive thermal lapse rate. Above the inflection point in the temperature profile called the tropopause, the lapse rate is negative in the region called the stratosphere. The source of heat in this region is the absorption of solar ultraviolet radiation by ozone. The stratopause (at about 50-55 km) marks the bottom of the mesosphere, a region of positive thermal lapse rate. Finally, above the mesopause, at roughly 85 km, is the thermosphere with negative lapse rate again produced by absorption of solar radiation. It is the mesosphere, also called the D region, which is of interest in this work.

The major constituents of the atmosphere are well mixed to an altitude

of approximately 105 km. That is, all species exhibit the same variation in concentration as a function of altitude. This property is taken as evidence of eddy diffusion, commonly characterized as turbulence. The altitude of 105 km is therefore called the turbopause. Above the turbopause, molecular diffusion dominates the mixing due to turbulence and the composition varies with altitude. Turbulence in the mesosphere is discussed at length in a subsequent chapter.

In addition to being well-mixed, the mesosphere is sufficiently ionized during the day by photoionization to be characterized as a weak plasma. The daytime electron concentration varies from approximately 10^2 electrons/cm^3 at 55 km to about 10^4 electrons/cm^3 at 90 km. In the same altitude range, the neutral molecule concentration varies from 10^{16}/cm^3 to 10^{14}/cm^3. The ratio of electrons to neutrals is therefore between 10^{-14} and 10^{-10} and the ionization has no effect on the motions of the neutral atmosphere. Recombination quickly removes the ionization after sunset, giving rise to radically different propagation conditions for daytime and nighttime communications with medium wave and shortwave signals.

Finally, the mesosphere is a dynamic region with a range of atmospheric motions. The atmosphere can be thought of as a compressible fluid, the important characteristics of which depend on the time scale of the motion of interest. For one class of waves, called gravity waves, the kinetic energy of the wave is exchanged with potential energy to form buoyancy oscillations. At shorter time scales, acoustic waves exchange kinetic energy with the compressional energy of the air. Motion at the shortest time scales involves the transfer of wave energy into thermal energy via turbulence. These motions are actually not distinct classes but rather represent a continuum of possible scales of motion. Acoustic-gravity waves are briefly

reviewed below. The importance of turbulence, both as a necessity for the coherent-scatter radar to operate and as a major topic of this work, merits a more lengthy treatment in a subsequent chapter.

1.3 Internal Acoustic-Gravity Waves

A general analysis of wave motions in the atmosphere is beyond the scope of this work. To obtain equations pertinent to gravity waves in the mesosphere, a simplifying assumption is employed: time scales longer than several hours and horizontal wavelengths of longer than 500 km are disallowed. This restriction effectively eliminates the spherical shape of the earth and the earth's rotation as significant factors. Tidal motions, which are therefore excluded, exhibit many of the same characteristics as gravity waves. The two-dimensional analysis summarized below has been carried out by Beer [1974] and others.

A coordinate system is chosen with x the direction of the horizontal propagation of phase and z the vertical direction. The corresponding velocities are U_x and U_z. Pressure and density are given by p and ρ, respectively. In order to employ a linear analysis, each variable is taken as a sum of a background value denoted by subscript o and a perturbation value shown with a subscript 1. All products of perturbation values are dropped.

The analysis is first carried out for a flat, isothermal, windless atmosphere, i.e.,:

$$U_{ox} = U_{oz} = \frac{\partial T}{\partial z} = \frac{\partial T}{\partial x} + \frac{\partial \rho_o}{\partial x} = 0 \tag{1.1}$$

where T is the temperature. The two components of the equation of motion, followed by the adiabatic equation and the mass continuity equation written in first order perturbed form are:

$$\rho_o \frac{\partial U_{1x}}{\partial t} = \frac{-\partial p_1}{\partial x} \tag{1.2}$$

$$\rho_o \frac{\partial U_{1z}}{\partial t} = \frac{-\partial p_1}{\partial z} - g\rho_1 \tag{1.3}$$

$$\frac{\partial p_1}{\partial t} + U_{1z} \frac{\partial p_o}{\partial z} = c^2\left(\frac{\partial \rho_1}{\partial t} + U_{1z} \frac{\partial \rho_o}{\partial z}\right) \tag{1.4}$$

$$\frac{\partial \rho_1}{\partial t} + U_{1z} \frac{\partial \rho_o}{\partial z} + \rho_o\left(\frac{\partial U_{1x}}{\partial x} + \frac{\partial U_{1z}}{\partial z}\right) = 0 \tag{1.5}$$

These equations are solved in the 4 variables p_1/p_o, ρ_1/ρ_o, U_{1x} and U_{1z} with the assumed solution of the form $A\exp[i(\omega t - K_x x - K_z z)]$ where K_x and K_z are complex wave numbers. The dispersion relation for these equations is therefore:

$$\omega^4 - \omega^2 c^2 (K_x^{\ 2} + K_z^{\ 2}) + (\gamma-1) g^2 K_x^{\ 2} + i\omega^2 \gamma g K_z = 0 \tag{1.6}$$

where c is the speed of sound, γ the ratio of specific heats and $c^2 = \gamma p/\rho$ has been used.

It is not possible in Equation (1.6) for K_x and K_z to be both real and non-zero. A change in wave amplitude will occur in that direction of K which has an imaginary part. Assume that $K_x = k_x$, i.e., it is entirely real. Then there is no amplitude variation in the horizontal direction. The dispersion relation then can be divided into its real and imaginary parts to yield the following conclusion: $Re(K_z) = 0$ or $Im(K_z) = \gamma g/2c^2 = 1/2H$ where $H = c^2/\gamma g$ is the scale height for this isothermal atmosphere.

The first alternative above means that the vertical wave number is entirely imaginary and therefore there is no variation of phase with height but rather an exponential amplification or decay. Both surface waves and evanescent waves fall into this category called external waves. The second

condition above, $\mathrm{Im}(K_z) = 1/2H$, yields internal wave solutions.

The dispersion relation for internal waves becomes

$$\omega^4 - \omega^2 c^2 (k_x^2 + k_z^2) + (\gamma-1) g^2 k_x^2 - \gamma^2 g^2 \omega^2 / 4c^2 = 0 \tag{1.7}$$

For any pair of k_x and k_z there are two distinct values for ω^2 and hence for ω if positive roots are used. Rewriting the dispersion relationship in the form

$$\frac{\omega^2 - (\gamma g/2c)^2}{c^2} + \left(\frac{(\gamma-1)g^2}{c^2 \omega^2} - 1 \right) k_x^2 = k_z^2 \tag{1.8}$$

it is clear that at least one term on the left side must be positive. Hence either $\omega^2 > (\gamma g/2c)^2$ or $\omega^2 < (\gamma-1)g^2/c^2$. The value of γ is necessarily less than 2 so that $(\gamma g/2c)^2 > (\gamma-1)g^2/c^2$. The conclusion is that a range of values for ω is not allowed under the assumption of internal waves.

Consider the case with $\omega^2 \gg (\gamma g/2c)^2$. Then the dispersion relation becomes $\omega^2 \simeq c^2(k_x^2 + k_z^2)$, the usual relation for sound propagation. The quantity $\gamma g/2c = \omega_a$ is called the acoustic cutoff frequency and the waves with $\omega > \omega_a$ are internal acoustic waves. Note that gravity still produces an exponential amplitude factor (the original condition for internal waves) even at frequencies much higher than ω_a.

For very low frequency waves, i.e., $\omega^2 \ll (\gamma-1)g^2/c^2$, the dispersion relationship becomes $\omega^2 \simeq (k_x^2/k_z^2)[(\gamma-1)g^2/c^2]$. The quantity $(\gamma-1)g^2/c^2 = \omega_g^2$ is called the buoyancy frequency or the Brunt-Vaisala frequency. It is the natural frequency of an atmosphere in hydrostatic equilibrium. Those waves with $\omega < \omega_g$ are therefore called internal gravity waves. A diagram of the allowable solutions to the dispersion relationship is shown in Figure 1.1.

The isothermal, windless model employed above does not adequately describe the mesosphere. The presence of tides in the mesosphere, although

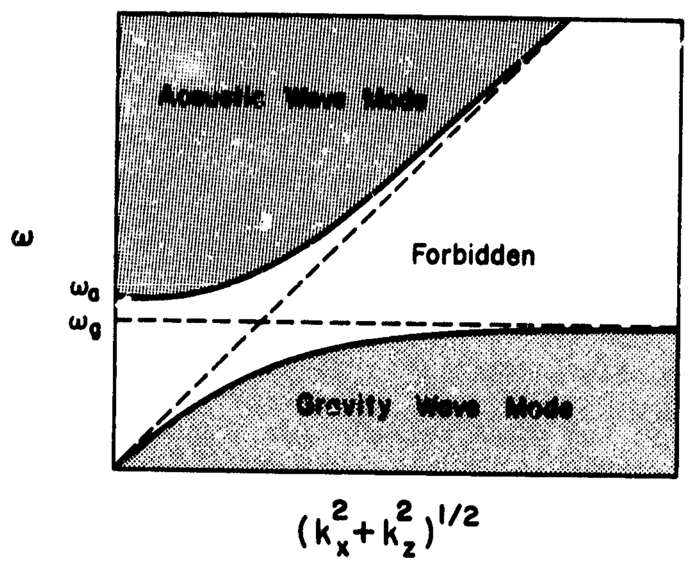


Figure 1.1 Plot of the dispersion relation for acoustic-gravity waves. (from Whitten and Poppoff 1971)

not of direct importance in this work, can provide a background wind profile for example. In the following analysis, a constant wind in the direction of phase propagation is assumed. This analysis provides an introduction to the more general case of a background wind shear, which will be discussed later.

The horizontal wind is taken as U_{ox} equal to a constant and the unperturbed temperature and density have no horizontal variation. By defining the operator

$$\hat{Q} = \frac{\partial}{\partial t} + U_{ox}\frac{\partial}{\partial x} = \frac{D}{Dt} - U_{1x}\frac{\partial}{\partial x} - U_{1z}\frac{\partial}{\partial z} \qquad (1.9)$$

one can write

$$\hat{Q}\,U_{1x} = \frac{DU_{1x}}{Dt} \qquad \hat{Q}\,U_{1z} = \frac{DU_{1z}}{Dt} \qquad (1.10)$$

The four required equations (x and z momentum, continuity, and adiabatic) become

$$\hat{Q}\,U_{1x} = \frac{-RT}{M}\frac{\partial}{\partial x}\left(\frac{p_1}{p_o}\right) \qquad (1.11)$$

$$\hat{Q}\,U_{1z} = -g\left(\frac{\rho_1}{\rho_o} - \frac{p_1}{p_o}\right) - \frac{RT}{M}\frac{\partial(p_1/p_o)}{\partial z} \qquad (1.12)$$

$$\hat{Q}\left(\frac{\rho_1}{\rho_o}\right) + \frac{\partial U_{1x}}{\partial x} + \frac{\partial U_{1z}}{\partial z} - \left(\frac{gM}{RT} + \frac{1}{T}\frac{\partial T}{\partial z}\right)U_{1z} = 0 \qquad (1.13)$$

$$\hat{Q}\left(\frac{p_1}{p_o}\right) + \gamma\left(\frac{\partial U_{1x}}{\partial x} + \frac{\partial U_{1z}}{\partial z}\right) - \frac{gM}{RT}U_{1z} = 0 \qquad (1.14)$$

where R is the gas constant, T the temperature and M the mean molecular mass [Beer 1974].

An exponential variation for the perturbed variables is assumed, e.g., $U_{1z} = W(z)\exp[i(\omega t - k_x x)]$ where $K_x = k_x$ has again been taken as entirely real (no horizontal variation in wave amplitude). The operator becomes

$$\hat{Q} = i(\omega - U_{ox}k_x) = i\,\Omega \qquad (1.15)$$

where Ω is taken as a Doppler frequency, i.e., the frequency relative to the background motion. The four equations then yield for W

$$\frac{d^2W}{dz^2} - \frac{1}{H}\left[1 + \frac{k_x^2c^2}{k_x^2c^2 - \Omega^2}\frac{dH}{dz}\right]\frac{dW}{dz} +$$

$$\left[\frac{k_x^2}{\Omega^2}\left\{\frac{(\gamma-1)g^2}{c^2} + \frac{g}{H}\frac{dH}{dz}\right\} + \frac{\Omega^2}{c^2} - k_x^2 + \frac{k_x^2c^2}{k_x^2c^2 - \Omega^2}\frac{1}{\gamma H^2}\frac{dH}{dz}\right] W = 0 \quad (1.16)$$

Solving this equation under the assumption that the atmosphere is slowly varying, leads to the conclusion that the isothermal dispersion relationship, Equation (1.6), can be used if the following substitutions are made:

$$\omega \rightarrow \Omega = \omega - U_{ox}k_x$$

$$\omega_g^2 \rightarrow \omega_B^2 = \omega_g^2 + \frac{g}{T}\frac{\partial T}{\partial z}$$

$$\omega_a^2 \rightarrow \omega_{an}^2 = \omega_a^2 + \frac{\gamma g}{2T}\frac{\partial T}{\partial z}$$

The first substitution is the Doppler shifted frequency mentioned earlier. The other two values are the non-isothermal values for the Brunt-Vaisala frequency and the acoustic cutoff frequency. Note that it is possible for ω_B^2 to exceed ω_{an}^2. For a mean molecular mass M = 29 and γ = 7/5, one has $\omega_B^2 > \omega_{an}^2$ if $\partial T/\partial z > 7.3$K/km. Recall that the lapse rate is positive in the mesosphere so that in general $\omega_{an} > \omega_B$.

1.4 Critical Layers

Consider Equation (1.6) for the special case $\Omega = 0$ while $\omega \neq 0$. That is, the Doppler shifted frequency is zero as a result of the background horizontal trace velocity in the direction of the phase propagation being equal to the phase speed. The coefficient of the W term approaches ∞ as $\Omega \rightarrow 0$ so that $W \rightarrow 0$. The perturbation amplitude is approaching zero so one concludes that the wave is absorbed.

The general case of a wind shear has been considered by Hines and Reddy [1967] and Booker and Bretherton [1967]. They also conclude that for $\Omega = 0$ absorption of the wave occurs. The altitude in the shear flow at which $\Omega = 0$ is called the critical level, and the surrounding region in which the absorption takes place, the critical layer. Booker and Bretherton [1967] assume a non-viscous model atmosphere so that turbulence is not generated but rather energy is transferred directly to the mean flow. Later work by Geller et al. [1975] and Fritts [1979], for example, show the generation of turbulence and radiating waves at a gravity wave-critical layer interaction. This work is discussed in greater detail in a subsequent chapter.

1.5 Objectives and Scope of the Investigation

The three principal objectives of this study are: (1) to review the theory of turbulent scatter and suggested sources for this turbulence, (2) to discuss the general characteristics of the radar data obtained at Urbana over several years, and (3) to examine in detail the spectra of the scattered signal in order to relate it to the sources of turbulence. An outline of this study is given below.

In Chapter 2 the theory of turbulent scatter is reviewed. Chapter 3 discusses the generation of this turbulence. The coherent-scatter experimental technique and the interpretation of spectra are discussed in Chapter 4. The experiments specific to this work and the associated data analysis are described in Chapter 5. A general description of the data from Urbana is given in Chapter 6. Detailed analysis of the echo spectrum is provided in Chapter 7. Conclusions and suggestions for future research appear in Chapter 8.

2. TURBULENT SCATTER THEORY

2.1 Introduction

The propagation of radio waves through the ionosphere is largely a function of the wavelength of the radio wave. The ionosphere is a weakly ionized plasma characterized by an electron plasma wavelength $\lambda_{pe} = (\pi/r_e N)^{1/2}$ where the magnetic field has been neglected, $r_e = 2.818 \times 10^{-15}$ m is the classical electron radius and N is the concentration of electrons. When a vertically propagating radio wave approaches an altitude where the plasma wavelength (which varies inversely with the changing electron concentration) and the radio wavelength are equal, the wave is reflected. Measurement of electron concentration using an ionosonde and the propagation of short-wave signals make use of this property.

The minimum plasma wavelength in the ionosphere is on the order of tens of meters. Radio waves with frequencies in the VHF region and above are therefore largely unaffected by the ionosphere. Small fluctuations in the plasma, however, can produce a weak but observable scattering of VHF signals. In the mesosphere these fluctuations are the result of the motions of the neutral atmosphere and the thermal motion of the individual components of the ionosphere. The scattering produced by the thermal fluctuations is referred to as incoherent scattering. In an incoherent-scatter experiment the autocovariance of the scattered signal is measured and compared to the autocovariance of signals scattered from laboratory plasmas to deduce ion and electron temperature, electron density and ion composition [Evans 1969]. Incoherent scatter is widely used for studies of the F and upper E regions where the plasma is not collision-dominated. In the D region and lower E regions of the ionosphere the frequent collisions of charged particles with neutral particles prevent the geomagnetic

field from having a large effect on the motion of the charged particles, force the neutral and charged species to have essentially the same temperature, and furthermore, imply that the variations in the ionization are tied to the motions of the neutral atmosphere. Specifically, the presence of tides, gravity waves and winds in the mesosphere can lead to regions of local instability where turbulence is generated. The motion of the neutrals in these turbulent eddies forces fluctuations at similar scale sizes in the ionization which in turn produces turbulent scatter. The Urbana radar which operates at a wave number corresponding to these scale sizes obtains echoes dominated by turbulent scatter. At smaller scale sizes the turbulent eddies are damped by viscosity and the returned signal for UHF radars is therefore dominated by incoherent scattering.

The relationship between the variations in ionization and the scattered signal observed by a coherent-scatter radar is explained by turbulent scatter theory. The mechanism of turbulent scatter was originally used to explain over-the-horizon tropospheric radio propagation [Booker and Gordon 1950]. In that case the scattering volume is determined by the intersection of the receiving and transmitting beams. For a monostatic pulsed radar the scattering volume is determined in a horizontal direction by the shared transmitting and receiving beam and in the vertical direction by the convolution of the region illuminated by the transmitted pulse and the region strobed by the range gate. If within the scattering volume the turbulence produces variations in ionization with spatial wavelength equal to 1/2 of the radar wavelength (i.e., the Bragg wavelength for a monostatic system) then the scattering from these variations arrives in phase at the radar. This is the basis for the name coherent-scatter radar. A brief review of turbulent-scatter theory as applied to coherent-scatter radar is

given below.

2.2 The Occurrence of Turbulence

Turbulence occurs in a fluid whenever certain conditions for stability are violated. In general, the motion of an incompressible fluid is governed by the continuity equation

$$\nabla \cdot \vec{V} = 0 \tag{2.1}$$

and by the equation of motion as given by the Navier-Stokes equation for the velocity field $V(x,t)$:

$$\{\frac{\partial \vec{V}}{\partial t} + \vec{V} \cdot \nabla\}\overline{V} - \nu\nabla^2\vec{V} + \frac{\nabla p}{\rho} = 0 \tag{2.2}$$

where p, ρ, V are the pressure, density, and velocity in the fluid and ν is the kinematic viscosity related to μ, the coefficient of viscosity, by $\nu = \mu/\rho$. The characteristic linear dimension L and the representative velocity V for a fluid flow are used to define a dimensionless parameter Re, the Reynolds number, given by

$$Re = \frac{VL}{\nu} \tag{2.3}$$

Although the specific critical value for Re above which a particular flow becomes unstable must be determined experimentally, in general Re >> 1 for turbulence to occur.

Only under certain conditions is turbulence energetically possible. The Brunt-Vaisala frequency ω_B represents a measure of the static stability of the fluid while $\partial V_o/\partial Z_o$, the gradient of the velocity in a direction perpendicular to stratification of the fluid, represents a perturbing effect tending to cause turbulence. The Richardson number

$$Ri = \omega_B{}^2/(\partial V_o/\partial z)^2 \tag{2.4}$$

is therefore a measure of the stability of a fluid flow. If the Richardson

number falls below some critical value then more energy is extracted from the mean flow than can be maintained by buoyancy oscillations. The excess energy is available for turbulent motion. If Ri < 0 the local ω_B is negative and buoyancy oscillations are not possible. The fluid is convectively unstable in this case and all the energy extracted from the mean flow is available for turbulence.

The critical value for the Richardson number, Ri_{cr} below which the onset of turbulence occurs is less than one and generally taken to be 0.25 [e.g., Drazin 1958; Breeding 1972; Einaudi and Lalas 1973; Geller et al., 1975]. Miles [1961] and Howard [1961] carried out a stability analysis to show that $Ri_{cr} = 0.25$. The value of Ri for the maintainance of a turbulent flow which already exists is Ri = 1.

2.3 Small-Scale Structure of Turbulence

A detailed study of the small-scale structure of turbulence, such as that contained in Hill and Bowhill [1976] is beyond the scope of this work. A summary of the important aspects of the theory of homogeneous turbulence, as detailed by Batchelor [1953] and Hinze [1959] is given below.

An important idea in the study of turbulence is that of similarity theory. First introduced by Kolmogorov, this theory basically asserts that turbulence is essentially the same at small length scales in all turbulent flows independent of the exact character of the flow at large length scales. (That is, if the various parameters of the flow are nondimensionalized then the measurements of data for turbulence generated by any method will fall on a single curve for small scale sizes.) The results of experiments in wind tunnels can therefore be extended to the ionosphere.

The similarity theory is only valid however when a wide range of length scales exists between the scale at which the turbulence is generated and the

scale at which it is dissipated to heat by viscosity. For flows with a high Reynolds number, such as those found in the atmosphere, this requirement is met. The nonlinear nature of the Navier-Stokes equation causes the turbulent energy generated at some large length scale to propagate through smaller and smaller length scales until it is eventually dissipated. The inhomogeneous method of generation of the turbulence is not discernible at the smallest scale sizes if the energy must pass through a wide range of scales. The small-scale turbulence for a flow with high Reynolds number is therefore expected to be locally homogeneous and isotropic [Batchelor 1953].

Kolmogorov further assumed that for small scales the structure of the turbulence would depend only on ε the eddy diffusion coefficient, the viscosity ν and the wave number k. Two important ranges of scales for the energy spectrum of turbulence $E(\overline{k})$ can be found. In the inertial subrange the viscosity has no effect on the spectrum, while at higher wave numbers the viscous subrange occurs where both ε and ν are important. The scale size which is characteristic of the viscous subrange is referred to as the Kolmogorov microscale or the inner scale. The scale size at which the turbulence is generated is correspondingly referred to as the outer scale.

A scalar field such as electron density which is being acted upon by a turbulent flow exhibits a scalar spectrum $\phi(\overline{k})$. The straining of the fluid by the turbulence tends to lengthen linear dimensions and stretch surfaces so that spherical perturbations become elongated. The straining is therefore responsible for the cascade of energy through smaller and smaller scale sizes. As a dimension of a perturbation in a scalar field becomes elongated the gradient increases and diffusion also increases. The diffusion however acts to spread out the perturbation. As the scale decreases the diffusion

eventually dominates the straining due to turbulence. The scalar spectrum therefore exhibits two types of ranges of wave numbers: a convective range where the convective effects on existing variations are important and a diffusive range where diffusion dominates.

The scalar spectrum is driven by $E(\bar{k})$ however, so the shape of $\phi(\bar{k})$ also depends on whether the inertial or viscous range applies for $E(\bar{k})$. If the diffusivity is much larger than the viscosity, then for increasing k the scalar spectrum moves from a convective range to a diffusive range while $E(\bar{k})$ remains in the inertial range. $\phi(\bar{k})$ is then said to exhibit an inertial-convective range followed by an inertial-diffusive range for increasing k. The ratio of the viscosity to the diffusivity, called the Schmidt number $Sc = \nu/D$, is less than 1 in this case. For $Sc >> 1$ the wave number increases to a very large value, well into the viscous range for $E(\bar{k})$ before $\phi(\bar{k})$ exhibits a diffusive range. The scalar spectrum therefore has three ranges: the inertial-convective range, the viscous-convective range and finally for large k (small scale size) a viscous-diffusive range. The wave-number dependence of both $E(\bar{k})$ and $\phi(\bar{k})$ are determined by dimensional analysis. A more detailed description of the extension to scalar fields is given by Hill and Bowhill [1976].

2.4 Scattering from Statistically Homogeneous and Stationary Medium

Turbulence is random both in space and in time: therefore, the variations in the ionization must be characterized as a random function of space and time. These random variations in ionization are equivalent to variations in the permittivity. Following the development of Rastogi and Bowhill [1976a] a permittivity relative to free space is taken as a random function of position $\bar{r}$ and time t with an average value and weak fluctuations; i.e., $\varepsilon(\bar{r},t) = \varepsilon_o(\bar{r},t) + \delta_\varepsilon(\bar{r},t)$ with $|\delta_\varepsilon(\bar{r},t)| << \varepsilon_o(\bar{r},t)$. This

random function is assumed to be statistically homogeneous and stationary with variations occurring slowly. With the scattering volume in the far field of the antenna the Born approximation is used and an expression for the scattered electric field is obtained. The autocovariance of the scattered field can be represented as

$$\int_V d^3\overline{r} \int_V d^3\overline{r''} \cdot \exp[i\overline{r''} \cdot \overline{K}_s] < \delta_\varepsilon(\overline{r},t)\ \delta_\varepsilon(\overline{r}-\overline{r''},\ t+\tau) \tag{2.5}$$

where v = the scattering volume

$\overline{r}$ = the vector distance of an arbitrary scattering element scattering volume

$\overline{k}_s = \overline{k}_i - \overline{k}_r$ = Bragg vector for a given incident $\overline{k}_i$ propagation vector and $\overline{k}_r$ scattering propagation vector

In this form the inner integral in equation (2.5) is recognized as a spatial Fourier transform of the autocovariance of the permittivity fluctuations at wave number $\overline{k}_s$. The transform is actually truncated because of the finite scattering volume so that contributions from wave numbers near $\overline{k}_s$ occur. The outer integral is simply an integral over the scattering volume.

The Bragg propagation vector for a monostatic radar is $\overline{k}_s = 2\overline{k}_i$ since $\overline{k}_i = -\overline{k}_r$. With wave number inversely proportional to wavelength equation (2.5) shows that variations in permittivity with spatial wavelength very near to half the radar wavelength produce the greatest part of the scattered field. The autocovariance of the electron-density fluctuations and the autocovariance of the permittivity fluctuations are directly related in the mesosphere by

$$R_{\delta\varepsilon}(\overline{r},\tau) = (\lambda_o{}^2 r_e/\pi)^2\ R_{\delta N}(\overline{r},\tau) \tag{2.6}$$

where r_e is the classical electron radius and λ_o is the radar wavelength.

The scattered field therefore can also be thought of as a direct consequence of variations in the electron density at wavelengths in the vicinity of the of the Bragg wavelength. The exact contribution at various wavelengths is determined by the pulse shape and the range gate convolved together as well as by the gain pattern of the antenna.

2.5 Scattering from a Locally Homogeneous and Locally Stationary Medium

In the previous section the scattered field was independent of time and spatial coordinate by virtue of the initial assumptions for the permittivity fluctuations. The ionosphere is not statistically stationary or homogeneous however: in fact the measurement of the variation in the statistical properties of the scattering volume represents the focus of the coherent-scatter experiment. The fluctuations in permittivity are therefore now assumed to yield a covariance function $R(\bar{r}_1,t_1,\bar{r}_2,t_2)$ with a corresponding power spectrum $S(\bar{k}_1,\omega_1,\bar{k}_2,\omega_2)$. Under the assumptions of local homogeneity and local stationarity the following representation is made:

$$R(\bar{r}_1 t_1, \bar{r}_2,t_2) = R_1\{\tfrac{1}{2}(\bar{r}_1+\bar{r}_2),\tfrac{1}{2}(t_1+t_2)\} \times \rho_2(\bar{r}_1-\bar{r}_2,\ t_1-t_2) \quad (2.7)$$

where R_1 is a non-negative function and ρ_2 is a statistically homogeneous and stationary correlation function. This allows the local changes of permittivity with time and space to be incorporated while retaining the concept of statistically stationary and homogeneous permittivity. The function R_1 can be regarded as the mean square value of the permittivity which is varying with space and time while ρ_2 has a non-varying shape for variations in space and time. Similarly:

$$S(\bar{k}_1,\omega_1,\bar{k}_1,\omega_1) = S_1\{\tfrac{1}{2}(\bar{k}_1+\bar{k}_2),\ \tfrac{1}{2}(\omega_1+\omega_2)\} \times \phi_2(\bar{k}_1-\bar{k}_2,\ \omega_1-\omega_2) \quad (2.8)$$

where $S_1(\bar{k}_1,\omega) = F_4[\rho_2(\bar{r},t)]$, $\phi_2(\bar{k},\omega) = F_4[R_1(\bar{r},\tau)]$

and F_4 denotes the 4-dimensional Fourier transform [Rastogi and Bowhill 1976a].

The received signal and its time covariance $R_e(t_1,t_2)$ can be evaluated in a way similar to that for the statistically homogeneous and stationary case. If a weighting function $W(\bar{r})$ and corresponding spatial Fourier transform $W(\bar{k})$ are defined to allow for the finite scattering volume then the expression for $R_e(t_1,t_2)$ can be seen to be the convolution of the weighting function and the transform of the covariance of the permittivity fluctuations:

$$R_e(t_1,t_2) = |X|^2 \{S_1(\bar{k}_o,\tau)\phi_2(0,t)\} \times |W(\bar{k}_o)|^2 \quad (2.9)$$

where $|X|$ is a constant dependent on system parameters, the variable transformations $t_1-t_2 = \tau$, $(t_1+t_2)/2 = t$ have been made, and $\bar{k}_o$ is the Bragg vector at the origin in the scattering volume. The weighting function is defined in a small neighborhood $\Delta\bar{k}$ (approximately equal to the inverse of the linear dimension of the scattering volume) of $\bar{k}_o$.

The permittivity fluctuations have been assumed to be locally stationary so the received signal is also assumed to be a locally stationary random process:

$$R_e(t_1,t_2) = R_e(t,\tau) = R_e'(t)\,\rho_e''(\tau) \quad (2.10)$$

where R_e' is a non-negative function and ρ_e'' is a statistically stationary random process. Equation (2.9) is therefore the product of a t-dependent term and a τ-dependent term:

$$R_e'(t) = \phi_2(0,t) \times |W(\bar{k}_o)|^2 \quad (2.11)$$

and

$$\rho_e''(\tau) = |X|^2\, S_1(\bar{k}_o,\tau) \times |W(\bar{k}_o)|^2 \quad (2.12)$$

when $\bar{k}_o >> \Delta\bar{k}$ or equivalently when the radar wavelength is a small fraction

of the dimensions of the scattering volume. Equation (2.12) is the result obtained previously for statistically stationary and homogeneous fluctuations. Again the instantaneous behavior of the covariance of the received signal is controlled by fluctuations at scales near the Bragg wavelength. The t-dependent term, equation (2.11), represents the contribution of fluctuations at scales comparable to the dimensions of the scattering volume.

A physical interpretation of the above might be as follows. Imagine the scattering volume to contain patches with permittivity fluctuations at the Bragg wavelength interspersed with regions where no fluctuations at the Bragg wavelength exist. The patches are created by turbulence within the scattering volume, destroyed by diffusion, or moved by winds and waves. The instantaneous variation in the covariance of the received signal corresponds to the amount of fluctuations, i.e., the energy in the spatial spectrum at the Bragg wavelength, within the patches. The long-term temporal changes in the spectrum correspond to changes in the orientation and speed of the patches in the scattering volume. This interpretation is discussed below.

2.6 Mixing-in-Gradient

The turbulence can be related to variations in electron density by consideration of the electron continuity equation:

$$\frac{\partial N}{\partial t} = q - L - \nabla \cdot (N\bar{v}) \qquad (2.13)$$

where q is the production term, L is the loss term, N is the electron density and $\nabla \cdot (N\bar{v})$ is the transport term. In general the production and loss term balance so that $q - L = 0$. The remaining term can be expressed as the sum of two terms $N \nabla \cdot \bar{v}$ and $\bar{v} \cdot \nabla N$. The first term represents a contribution made possible by the compressibility of the atmosphere. The

second term describes the mixing-in-gradient process whereby the time-varying neutral velocity mixes the mostly vertical electron-density gradient. The scalar spectrum of the electron density fluctuations and therefore the permittivity fluctuations are coupled to the turbulence spectrum.

Rastogi and Bowhill [1976b] show that the VHF radar at Jicamarca, Peru transmitting at 40.92 MHz operates just above the Tchen range of wave numbers where the turbulent energy spectrum is making a transition from the inertial to the viscous subranges. The authors conclude that in this range an increase in the energy dissipation rate produces an increase in the radar cross section. The variations in returned signal intensity are therefore attributed to the spatial and temporal intermittency of turbulence. Furthermore, the Doppler velocity of the neutrals is observed as a Doppler shift in the return signal so that the coherent-scatter radar can use the turbulence as a tracer independent of the source of the turbulence. The conclusions above also apply to the Urbana Radar operating at 40.92 MHz.

2.7 Reflection vs. Scattering

In addition to the mechanisms of Thomson scatter and turbulent scatter radar returns are also obtained by scattering from meteor trails and from a process known variously as partial, Fresnel or diffuse scattering. Meteoric scattering is of limited interest for this work: generally the scattering from meteors tends to contaminate the daytime coherent-scatter echoes with short bursts of power while the intermittency of meteors precludes their use on a regular basis for nighttime measurements. Examples of scattering from meteors are given below.

Partial reflection has been studied at frequencies below VHF for years [Gregory 1961; Weiland and Bowhill 1981]. Evidence for reflection from

the troposphere and stratosphere has recently been given by several authors [e.g., Gage and Green 1978; Rottger and Liu 1978; Rastogi and Rottger 1982]. A model for Fresnel scattering developed by Gage et al. [1981] shows good agreement with tropospheric data. Evidence for Fresnel reflection has been given for mesospheric heights also [Rottger et al., 1979; Fukao et al., 1979]. In general these authors believe that both reflection and scattering are observed by VHF radar. The development of the radar equation for these two cases is reviewed below.

To obtain the radar equation for Fresnel reflection a horizontally stratified atmosphere with variations in altitude of the refractive index n(z) is assumed. The received power for a monostatic radar is:

$$P_r = \frac{P_t G^2 \lambda^2}{16\pi^2 Z^2} |r|^2 \tag{2.14}$$

where P_t is the transmitter power, G the antenna gain and λ the wavelength. The reflection coefficient r is:

$$r = \int_{-\ell/2}^{+\ell/2} \frac{1}{2n} \frac{\partial n(z)}{\partial z} \exp(-jkz)dz \tag{2.15}$$

where $j = \sqrt{-1}$, $k = 4\pi/\lambda$ is the Bragg wave number and ℓ is the thickness of the reflecting structure [Rottger 1980]. Clearly the steepness of the gradient can have a large effect on the amount of reflection. It is important to note that different shapes of the n(z) profile can produce identical reflection so that identification of n(z) from P_r is not possible.

If turbulence produces variations in refractive index throughout the volume formed by the antenna pattern, the radar pulse and sampling gate, then under the assumptions of weak and single scattering the received signal is:

$$P_s = \frac{P_t G' \lambda^2 \Delta z}{16\pi^2 z^2} \eta \tag{2.16}$$

where ΔZ is the vertical extent of the scattering volume and n is the radar reflectivity. G' is an effective antenna gain which is based on the non-uniform illumination of the scattering volume by the antenna pattern. The radar reflectivity is related to the fluctuations in the refractive index by:

$$\eta = \frac{\pi^2}{2} k^4 F_\eta(\bar{k}) \tag{2.17}$$

where $F_\eta(k)$ is the one-dimensional spatial spectral density of the refractivity irregularities [Ottersten 1969; Tatarskii 1971]. If the turbulence is assumed to be in the inertial subrange and integration is performed over times longer than the correlation time of the turbulent structures then the radar reflectivity can be shown to be:

$$\eta = 0.39\, C_n^2 \lambda^{-1/3}$$

where

$$C_n^2 = 5.26\, \overline{\Delta n^2}\, L_o^{-2/3}$$

The quantity C_n^2 is referred to as the refractive index structure constant and represents a measure of the outer scale L_o and the mean square fluctuations of the refractive index [Rottger 1980]. Experimental methods for investigating the relative amounts of scattering versus reflection are discussed in Chapter 8.

3. SOURCES OF TURBULENCE

3.1 Introduction

The generation of turbulence in the lower atmosphere, particularly clear air turbulence (CAT) has been extensively studied [Gossard and Yeh 1980]. In this chapter several mechanisms responsible for the generation of turbulence are reviewed. Convective instability is shown in Section 3.2 to be the result of a super-adiabatic lapse rate. Kelvin-Helmholtz instability (KHI) or shear instability is the result of the relative motion of a fluid which is statically stable. KHI is discussed in Section 3.3. Gravity-wave breaking which is reviewed in Section 3.4 is the result of the increasing amplitude of a wave as it propagates upward. Local instabilities produced by the high amplitudes generate turbulence. Finally the critical layer generation of turbulence is discussed in Section 3.5. It is found that the critical layer is important in other methods of generation of turbulence.

3.2 Convective Instability

A region of the atmosphere is convectively unstable when a parcel of air displaced from its equilibrium position tends to continue moving in the direction of displacement rather than to return to its original position. In this case $\omega_B^2 < 0$ and gravity wave motion is not possible. The value of ω_B^2 is determined by the relationship between the actual temperature profile and the adiabatic lapse rate as described below.

The derivation of the equations for gravity-wave propagation in Chapter 1 gives a definition of the Brunt-Vaisala frequency as

$$\omega_B^2 = \frac{(\gamma-1)g^2}{c^2} + \frac{g}{T}\frac{\partial T}{\partial z} \tag{3.1}$$

where γ is the ratio of specific heats, c is the speed of sound, T is the temperature, z is the vertical coordinate and g is the acceleration due to

gravity. In order to relate this definition to the adiabatic lapse rate an alternate definition is given based on the potential temperature θ which is the temperature a parcel of dry air at pressure p and temperature T would have if it contracted or expanded adiabatically to a pressure of 1000 mb. That is:

$$\theta = T\left(\frac{1000}{p}\right)^{\frac{\gamma-1}{\gamma}} \tag{3.2}$$

The alternate definition of the Brunt-Vaisala frequency based on potential temperature is

$$\omega_B^{\ 2} = \frac{g}{\theta}\frac{d\theta}{dz} \tag{3.3}$$

The sign of $\omega_B^{\ 2}$ is therefore the sign of the vertical gradient of potential temperature.

The two definitions of $\omega_B^{\ 2}$ given above in equations (3.1) and (3.3) are equivalent for an atmosphere in hydrostatic equilibrium, i.e., for which

$$\frac{\partial p}{\partial g} = -\rho\ g \tag{3.4}$$

where ρ is the density. Performing the differentiation of equation (3.3) on the definition of potential temperature and factoring yields

$$\omega_B^{\ 2} = \frac{g}{T}\frac{\partial T}{\partial z} + \frac{g}{p}\left(\frac{1-\gamma}{\gamma}\right)\frac{\partial p}{\partial z} \tag{3.5}$$

Substitution of equation (3.4) and the identity

$$c^2 = \gamma p/\rho \tag{3.6}$$

yields equation (3.1), the original definition.

Lapse rate is defined to be positive when the temperature decreases with increasing altitude. The adiabatic lapse rate is therefore the negative of the vertical temperature gradient under the condition that the gradient of potential temperature is zero. Specifically:

$$-\left.\frac{\partial T}{\partial z}\right| = T\frac{(\gamma-1)g}{c^2} \qquad \frac{\partial \theta}{\partial z} = 0 \tag{3.7}$$

Clearly under this condition a parcel which is displaced adiabatically will be in equilibrium at its new position and will not be further accelerated. If the lapse rate is greater (i.e., $\partial T/\partial Z$ is more negative) then $\omega_B^2 < 0$ and convective instability occurs. As a gravity wave attempts to propagate through the unstable region parcels of air are accelerated away from their equilibrium positions and continue to accelerate because of the instability. Parcels moving upward force other nearby parcels to move down and take their place so that eddies are formed. Eventually the transfer of energy due to the turbulent eddies modifies the temperature profile and the region becomes convectively stable.

3.3 The Kelvin-Helmholtz Instability

The Kelvin-Helmholtz instability (KHI) is a dynamic rather than static instability. A KHI is produced in a stable, stratified fluid by the relative horizontal motion of adjacent layers in the fluid. Essentially the stabilizing influence of a decreasing density with increasing altitude is overcome by the concentration of vorticity which occurs in a region with high shear. The result is the development of a breaking wave pattern often referred to as cat's-eyes. Various aspects of KHI are discussed below.

The elimination of the perturbations of density, pressure and vertical velocity from the equations of motion in the presence of a background wind yields the Taylor-Goldstein equation for the perturbation in the horizontal direction:

$$(U-c)(\phi''-k_x^2\ ) - U''\phi + J\phi/(U-c) = 0 \qquad (3.8)$$

where z is the vertical coordinate, U is the horizontal mean velocity in the x direction but varying only with z, c is the phase speed of the wave and the prime denotes differentiation with respect to z. J is the Richardson number here, chosen so as to be in notational agreement with Drazin [1958]

and others. The stream function is

$$\psi' = \phi(z)\exp[ik_x(x-ct)] \tag{3.9}$$

with

$$u = \partial\psi'/\partial z \qquad w = -\partial\psi'/\partial x = -ik_x\psi' \tag{3.10}$$

where u and w are the horizontal and vertical perturbation velocities [Drazin 1958]. The phase speed c can be complex - a positive imaginary part represents an unstable growing wave. The two-dimensional approach is acceptable because perturbations transverse to the direction of the mean flow are stable [Chandrasekar 1961].

The appearance of J, the Richardson number, in the Taylor-Goldstein equation indicates that the value of J and the occurrence of instability are related. Chandrasekar [1961] argues that the source of KHI lies in the kinetic energy of the mean flow and that this must exceed the change in potential energy required to move a parcel of air against the pressure gradient. Equating the two energy values, Chandrasekar [1961] suggests that for $J > 1/4$ the fluid is stable. The work of Miles [1961] and Howard [1961] shows that sufficient conditions for the stability of a shear flow in a heterogeneous fluid are that the gradient of horizontal velocity $U'(z)$ be nonzero and that the Richardson number J be greater than 1/4. Furthermore, the complex wave velocity for any unstable mode must lie inside the semi-circle in the upper half-plane of c space which has the diameter given by the range of the horizontal velocity across the shear region [Howard 1961]. This requires that a critical level exist within the shear region for every unstable wave. KHI, therefore, involves the transfer of energy from the mean flow into wave energy near a critical level. The opposite case of a wave giving up energy to the mean flow at a critical level is discussed below.

Experimental observations and numerical simulations of KHI are numerous. Scotti and Corcos [1972] vary the Richardson number at small values of the Reynolds number and obtain a critical Richardson number of J = 0.22. Patniak et al., [1976] use a numerical simulation to study the development of KHI and the cat's-eye in particular. The cat's-eyes form very quickly in their model so that initial growth cannot be well observed. Once formed however, the narrow region of cat's-eyes widens and an increasing fraction of the vorticity which was initially spread throughout the shear region becomes concentrated in the cat's eyes. The cat's-eyes are referred to as cores and the regions of concentrated streamlines between the cores are referred to as braids. Eventually the cat's-eyes reach an equilibrium state and the width of the region stops growing. Patniak et al., [1976] also illustrate the generation of subharmonics and the subsequent widening of the cat's-eye region. Figure 3.1 shows the relative motions of the fluid in the cores and braids. Note the straining at the center of the braid. In this diagram the fluid is moving from left-to-right at the bottom and right-to-left at the top.

The finite thickness in the simulation of Patniak et al., [1976] is discussed by Corcos and Sherman [1976]. These authors indicate that vorticity is generated in the braids by the tilt of the braids; i.e., a larger tilt in the braids produces a larger vorticity because of the difference in density along the braid. The vorticity in the braids is drawn into the cores causing the cores to grow. Eventually the cores draw in as much vorticity as can be generated in the braids and the cores stop growing.

The generation of turbulence from KHI is discussed by several authors. Woods [1969] shows that for flows in the ocean with Reynolds number Re < 300 the turbulent final state of the instability is not present. Woods also

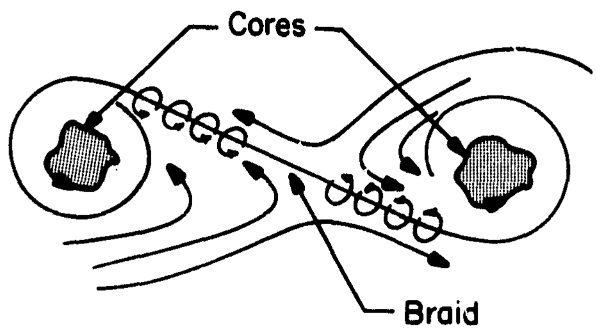


Figure 3.1 Illustration of core regions separated by a vorticity producing braid.

suggests that the turbulence which appears for higher Reynolds numbers in the initial flow is the result of secondary instabilities. From a laboratory study of liquids in a tank Thorpe [1973] suggests that convective instability may be responsible for the turbulent breakdown of KHI. Corcos and Sherman [1976] indicate that for high Reynolds number secondary instability is likely. However, their expe. .ent is limited to low Reynolds number and no turbulent breakdown is observed. Peltier et al. [1978] carry out a numerical simulation at Reynolds numbers both above and below the Re = 300 threshold observed by Woods. Again no secondary instabilities are observed but the authors suggest that convective instability could be realized in the superadiabatic regions above and below the cores. Furthermore the two-dimensional finite amplitude KHI may only be unstable to three-dimensional perturbations so that no secondary instability would be observed Peltier et al. [1978] also show that the mean shear profile and the density profile are split by the wave-mean flow interaction to produce new stable regions which each contain an inflection point in the modified shear flow. This may be the cause of the splitting of turbulent layers as observed by Atlas et al. [1970].

Additional studies of KHI have involved the numerical simulation of a tanh profile of velocity in the presence of a lower boundary (e.g, Lindzen and Rosenthal [1976]; Lalas and Einaudi [1976]; Davis and Peltier [1979]). In general these studies show the presence of modes which can propagate away from the shear layer and be reflected in the cavity between the ground and the shear. The Kelvin-Helmholtz waves or cat's-eyes however, are largely evanescent away from the shear layer.

Finally, the occurrence of KHI in the atmosphere has been observed with radar. Gossard and Richter [1970] and Atlas et al. [1970] show the develop-

ment of KHI using a 2 meter resolution with a vertically pointing FM-CW radar. Klostermeyer and Ruster [1980] compare the observations of a jet-stream-generated KHI with a model and find the two to be in good agreement. Clearly KHI is an important source of turbulence in the atmosphere.

3.4 Gravity-Wave Breaking

Gravity waves which propagate upward in a neutral atmosphere grow in amplitude with increasing height. As the pressure decreases with altitude the amplitude of the wave increases proportionally because the energy flux is constant. The increasing amplitude eventually produces regions of instability by modification of the background atmosphere and generation of small regions of high shear. The instabilities eventually lead to the generation of turbulence. This process is discussed below.

In the analysis of Chapter 1 the effect of a gravity wave on the local temperature and density was not considered. Hodges [1967] determines the perturbation of temperature and density with respect to horizontal velocity. A perturbed Brunt-Vaisala frequency is then calculated and finally a modified Richardson number is found. An expression for the local Richardson number in the gravity wave is of the form

$$J(x,z) \simeq \frac{1-A_1\sin\phi_1}{A_1^2\cos^2\phi_1} \qquad (3.11)$$

where x and z are the horizontal coordinate in the direction of propagation and the vertical coordinate, respectively. The angle ϕ_1 represents the phase within the gravity wave at (x,z) while A_1 is a complicated expression proportional to the horizontal velocity of the wave at (x,z). For values of $A_1 > 1$ there exist regions around odd multiples of $\pi/2$ for ϕ_1 in which $J < 0$. As discussed above this condition implies convective instability and the generation of turbulence. Hodges [1967] concludes that planar regions of

instability perpendicular to the direction of propagation of the gravity wave will propagate with the wave. The turbulence generated in these planar regions will be dissipated in the interspersed stable regions. The analysis of Hodges [1967] is carried out under the assumption that the perturbations in temperature and density remain small enough so that a linear solution is possible.

Using a numerical model Jones and Houghton [1972] allow nonlinear interaction of the wave with itself and with the background atmosphere. A wave packet is propagated upward through an atmosphere with zero mean wind. A substantial fraction of the wave energy is transferred directly to the mean flow to form a critical layer. Negative Richardson numbers, implying convective instability, are found at altitudes corresponding to the precursor of the wave packet. Jones and Houghton [1972] conclude that a combination of convective instability and critical-layer interaction occurs in the wave breaking process. Breeding [1972] shows the presence of both dynamic and static instability in his work. Again a critical layer is formed leading to the further generation of turbulence as discussed below.

3.5 Critical Layers

The critical level for a gravity wave in a shear flow is the level at which the horizontal phase speed of the wave and the horizontal velocity of the mean flow are equal [Bretherton 1966]. As shown in Chapter 1 of this work at a critical level the vertical perturbation velocity goes to zero and the wave can be absorbed. The transfer of momentum from the gravity wave into the mean flow can occur with or without the intermediate generation of turbulence. This process is discussed below.

In the analysis of Booker and Bretherton [1967] the Richardson number $J > 1/4$ at all times so that the generation of dynamic instability is not

possible. Furthermore, the effects of viscosity and heat transfer are ignored also. The transfer of the wave energy to the mean flow is not dissipative in this case, but rather the wave energy serves to directly modify the mean flow. The dispersion relationship given in Chapter 1 (equation 1.8) can be rewritten for the case of a background wind as follows:

$$\frac{\Omega^2 - \omega_{an}^{\;2}}{c^2} + k_x^{\;2}\left(\frac{\omega_B^{\;2}}{\Omega^2} - 1\right) = k_z^{\;2} \tag{3.12}$$

Recall that Ω, the frequency relative to the background wind goes to zero as the critical level is approached. Therefore $k_x^{\;2}$ must gradually go to zero implying an increasing horizontal wavelength. The vertical perturbation velocity vanishes and the horizontal perturbation, with infinite wavelength is indistinguishable from the mean flow.

A more detailed analysis [Bretherton 1966] shows that the wave does not reach the critical level in a finite length of time for an arbitrarily large Richardson number. If the Richardson number is finite then some wave motion does propagate through the critical layer but with an attenuation which is severe for a moderate value of the Richardson number [Booker and Bretherton 1967]. The critical layer therefore acts as a filter - an aspect of critical layers which is discussed by Hines and Reddy [1967]. Clearly, for a mean wind which changes with height, waves with different phase velocities reach a critical level at different altitudes. The spectrum of wave energy incident on the wind profile is filtered in this way since the spectrum is dependent on altitude.

Jones [1968] shows that the gravity wave critical-layer interaction for values of the Richardson number $J < 1/4$ is one of reflection rather than absorption. The wave can be over-reflected, i.e., the reflected wave is

larger in amplitude than the incident wave as the wave extracts energy and momentum from the mean flow. Geller et al. [1975] show that even for initial Richardson numbers greater than unity, regions of dynamic instability and of convective instability can be created in the vicinity of the critical layer. The regions of large vertical shear in the horizontal wave velocity are found to be on the order of 10 to 100 meters thick. The nonlinear model of Geller et al. [1975] shows the downward motion of the critical level but is not capable of describing the development of the unstable regions. Fritts [1978] employs a time-dependent nonlinear model to show the development of Kelvin-Helmholtz instabilities in the regions of high shear formed by the critical layer. As in Fritts and Geller [1976], Fritts [1978] finds that viscosity serves to stabilize the critical layer interaction. Nonlinear interactions between waves has a stabilizing effect only for low-viscosity flows. When the nonlinear or viscous effects are not sufficient to stabilize the interaction the Kelvin-Helmholtz instability grows until it dominates the flow. These instabilities are initially excited by the harmonics of the incident wave and grow by drawing energy from the unstable velocity shears [Fritt 1978]. In this manner the gravity wave-critical layer interaction leads to thin regions of turbulence.

It is clear that the critical layer generation of turbulence and the generation of turbulence due to a Kelvin-Helmholtz instability are related. The radiating and over-reflective modes observed for a KHI in the presence of a lower boundary [Davis and Peltier 1979] are similar to those observed for a critical layer interaction [Fritts 1979]. This is reasonable since Howard [1961] points out that every unstable mode for a KHI must have a critical level within the shear. Furthermore, Breeding [1972] shows the development of a critical layer due to the overamplification of upward propagating

gravity waves. The various mechanisms for turbulence generation are therefore interrelated. Klostermeyer [1980] discusses a model capable of showing the various interactions. However, analysis of the specific mechanism by which a particular turbulent layer in the mesosphere has originated remains a difficult task without a detailed history of the velocity field.

4. EXPERIMENTAL TECHNIQUE

4.1 Introduction

Coherent-scatter radar experiments observe the dynamics of the atmosphere by measurement of the spectra of returned signals. A scatterer moving with a velocity V along the line-of-sight of the radar produces an echo with a Doppler shift f_d given by

$$f_d = 2V/\lambda \tag{4.1}$$

where λ is the radar wavelength. For the Urbana radar which operates at 40.92 MHz a velocity of 5 m/s toward the radar (downward) produces a Doppler shift of 1.36 Hz. The radar is pulsed so that the range of the scatterer can be determined. Coherent-scatter radar is therefore an example of a pulse-Doppler radar as discussed by Skolnik [1962].

The antenna noise temperature expected at low-VHF frequencies is on the order of 10^4 K [Jordan and Balmain 1968]. Furthermore the scattering process is very weak so that a sensitive radar is required. A transmitter with peak pulse power on the order of 1 MW and an antenna aperture of thousands of square meters typically produce a signal-to-noise ratio at the output of the coherent detector of less than one. In order to improve the signal-to-noise ratio a signal-processing technique called coherent integration is employed. This process is described in Section 4.2.

The relationship of spectral parameters to atmospheric dynamics is discussed in Section 4.3 and used to interpret several examples of spectra in Section 4.4. One aspect of this interpretation is the correspondence between individual turbulent cells and discrete peaks in the spectrum. In Section 4.5 it is shown that for a spectrum with many peaks it is impossible to associate all of the peaks with individual regions of turbulence. A discussion of the experiments particular to this work is deferred to a later

chapter.

4.2 Coherent Integration

The presence of noise in a receiving system, either from the receiver itself or in the form of sky noise, places a limit on the detectability of signals. For a coherent-scatter radar such as that at Urbana even the strongest scattered returns produce a signal-to-noise ratio on the order of one. The process of coherent integration was first used by Woodman and Guillen [1974] to improve the signal-to-noise ratio so that the subsequent calculation of the spectrum could take place for data which would otherwise have been unusable. The process is fundamentally a recognition of the great difference in bandwidth between noise and signal in the samples. A description is given below.

The primary factor which determines the necessary receiver bandwidth is the height resolution of the radar. In order to achieve maximum signal-to-noise performance the receiver should be a matched filter for the transmitted pulse. If the receiver bandwidth is widened then noise is added since it is present at all frequencies in the passband. The signal bandwidth is that of the transmitted pulse with a very small Doppler shift - increasing the receiver bandwidth beyond this adds noise but no signal energy so that the signal-to-noise ratio is reduced. If the bandwidth is too narrow then both signal and noise are lost but the fraction of signal energy lost is always greater than the amount of noise energy lost. Again a loss in signal-to-noise ratio occurs. More importantly the loss of the high-frequency information implies correlation of the signal for a longer time. The time after transmission corresponds to range information for a pulsed radar so that introducing a correlation time at the receiver is equivalent to reducing the range resolution. The receiver is therefore a matched

filter to permit maximum possible range resolution while maintaining the best noise performance.

For each radar pulse a set of samples corresponding to different altitudes is collected. Since many radar pulses are transmitted during one minute a sequence of samples is obtained for each altitude and the spectrum is calculated from that sequence. The bandwidth of the signal portion of these samples is that of the Doppler frequencies, i.e., 4 Hz or less: the high-frequency information associated with the return from each pulse is removed by the sampling. The Nyquist sampling rate is therefore 8 Hz which requires that at least 8 transmitter pulses and associated samples must occur every second. The noise is uncorrelated with the signal and with the pulse repetition frequency of the radar so that the noise bandwidth is unaffected by the sampling. The process of coherent integration provides a low-pass filter (actually a comb filter) which approximates a matched filter for the samples.

If the radar is operated at a pulse-repetition frequency greater than 8 Hz, say 200 Hz, then the signal will be coherent (have the same phase) for the 25 samples at 200 Hz which will replace a single sample at 8 Hz. The noise will be uncorrelated from sample to sample however so that by integrating (adding) the 25 samples together the signal will add constructively but the noise will tend to add to zero. Therefore, the coherently integrated samples have improved signal-to-noise ratio. The factor by which the signal to-noise ratio is improved is equal to the number of samples added [Schmidt et al., 1979]. Adding too many samples together causes the effective sample rate of the coherently integrated samples to fall below the required Nyquist sampling frequency. The low-pass filter formed by the coherent-integration process has a cutoff frequency too low for the signal bandwidth in this

case. If too few samples are added together then the noise performance is not optimum and additional computation time is needed in the later processing stages. Clearly the process of coherent integration serves as a matched filter to improve the signal-to-noise ratio before calculation of the spectrum.

4.3 Obtaining Geophysical Information

For a typical coherent-scatter experiment the power, Doppler frequency and correlation time are measured on a minute-by-minute basis. These parameters can be obtained from either the power spectrum or the autocorrelation of the signal since these form a Fourier transform pair. Calculation of the power spectrum is done off-line [Schmidt et al., 1979] or by suspending data collection for a short period each minute [Carter et al., 1980]. The latter technique forces a reduction in signal-to-noise ratio because some of the available data each minute is not processed but has the advantage that an immediate visual access to the data is available. The autocorrelation is computationally more efficient and is therefore calculated in real time [Countryman and Bowhill 1979; Gibbs and Bowhill 1979]. The geophysical interpretations of the parameters power, Doppler frequency, and correlation time are given below.

In a well-designed radar the noise power should be independent of altitude as discussed above. Variations in power with altitude are therefore attributed to the presence of scattering or reflection at a particular altitude. The power is obtained from the area under the power spectrum curve or from the autocorrelation evaluated at lag zero. In either case the measured power is for signal plus noise. Signals are assumed to be present at only some altitudes so the minimum value of power is used as an approximate value for the noise power.

The relationship between Doppler frequency and the velocity of a target along the line-of-sight of the radar has been given in equation (4.1). If the line-of-sight of the radar is entirely vertical then the observed variation of velocity from minute-to-minute is primarily due to gravity-wave motion of turbulent cells. If the beam points obliquely then horizontal velocities associated with tides can produce a contribution to the line-of-sight velocity and hence to the measured Doppler frequency. The Doppler frequency is the centroid of the power spectrum or first derivative of the phase of the autocorrelation at lag zero.

The correlation time of the signal is generally taken to be the time required for the amplitude of the autocorrelation to fall to some specific fraction of its value at lag zero. The correlation time and the width of the spectrum are inversely related by virtue of the Fourier transform process. An interpretation of the correlation time is difficult because so many factors can contribute. If the signal is present only a short time during the period for which the spectrum has been calculated then the spectrum will be very wide or the correlation time correspondingly very short. When the signal is present throughout the measurement then the width of the spectrum is an indication of the character of the turbulence. Imagine the turbulent eddy at the outer scale having a large angular velocity. Then the velocities at opposite sides of the cell will be quite different so that a large spread in the spectrum occurs. If the turbulent cell spins slowly then the velocities will not vary much from the average and the spectrum is fairly narrow. Furthermore, multiple scattering regions within the scattering volume may have different velocities which spread the spectrum. Examples of the various kinds of spectra are given below.

4.4 Examples of Spectra

The spatial and temporal intermittence of turbulence in the mesosphere implies that a large fraction of the measured spectra show noise only. An example of a noise spectrum is shown in Figure 4.1. The date, start time, and altitude of this power spectrum of 64 seconds of data are given in the figure along with a number which is proportional to the maximum value of the linear scale of power. That is, all spectra are scaled so that the maximum value corresponds to the highest point on the power axis. Comparison of Figure 4.1 to those which are discussed below shows that for 64 seconds of data a scaling factor on the order of 10^7 indicates noise.

When a meteor passes through the beam of the antenna its ablation leaves a trail of metallic ions which can scatter the radar signal. The trail diffuses quickly so that the return from a meteor while very strong for a few seconds, rarely lasts a minute. The 64-second spectrum of a meteor return is shown in Figure 4.2. As expected, the spectrum is somewhat noisy and exhibits a wide spread in velocity because of the limited duration of the signal during the 64 second time period. The presence of such a signal in only a single minute of data surrounded by noise in the preceding and succeeding minutes at the same altitude is an indication of a meteor.

An 8-second power spectrum of a meteor return is shown in Figure 4.3. The fundamental difference in the appearance of this spectrum compared to Figure 4.2 is the reduced resolution in the frequency domain corresponding to the use of a smaller number of samples. The spread in velocity about the Doppler velocity is roughly the same in both figures, implying that the 8-second interval may also be too long a time period when dealing with echoes from meteors. The temporal behavior of meteor echoes observed by coherent-scatter radar is discussed in greater detail below.

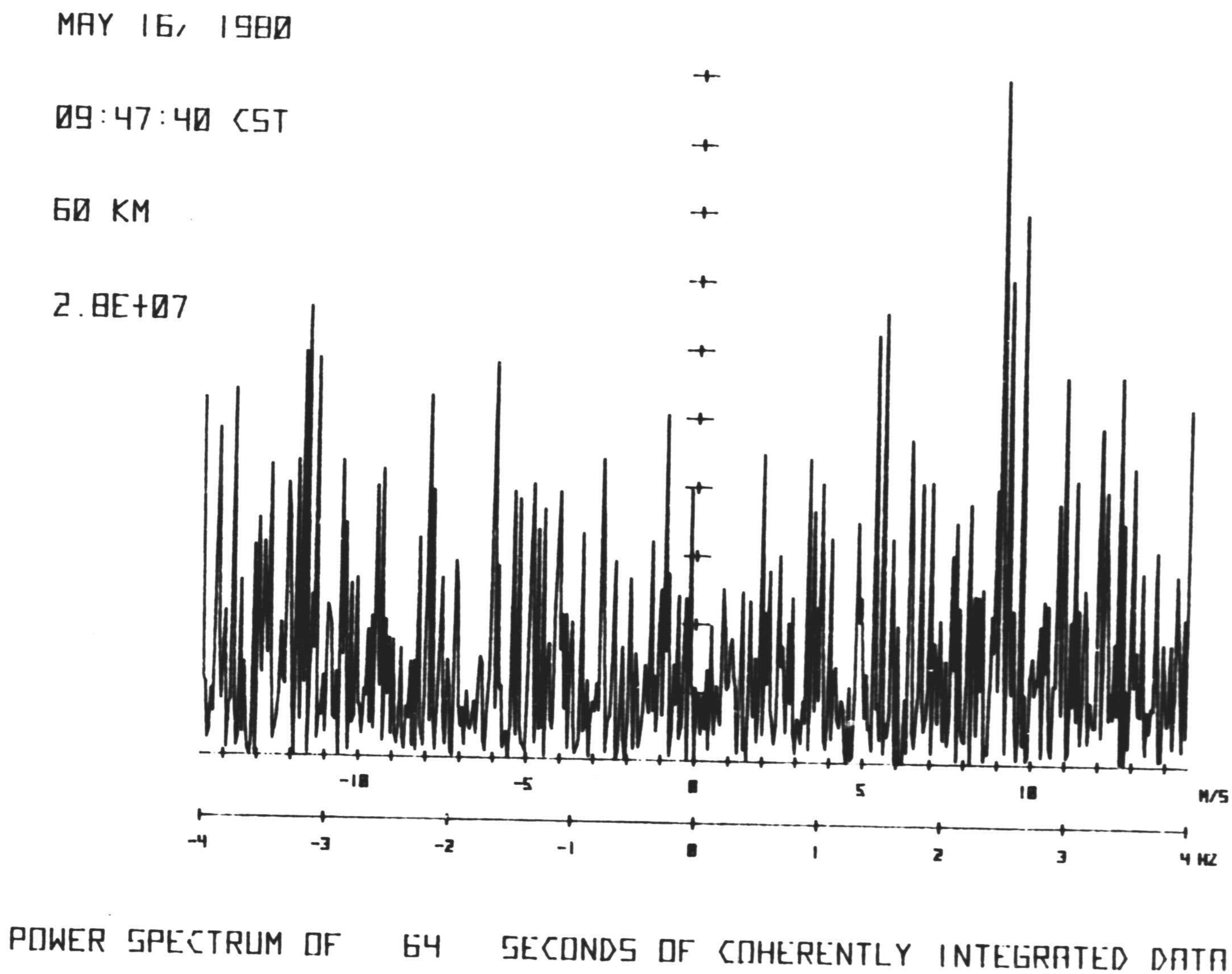


Figure 4.1 Power spectrum of 64 seconds of noise.

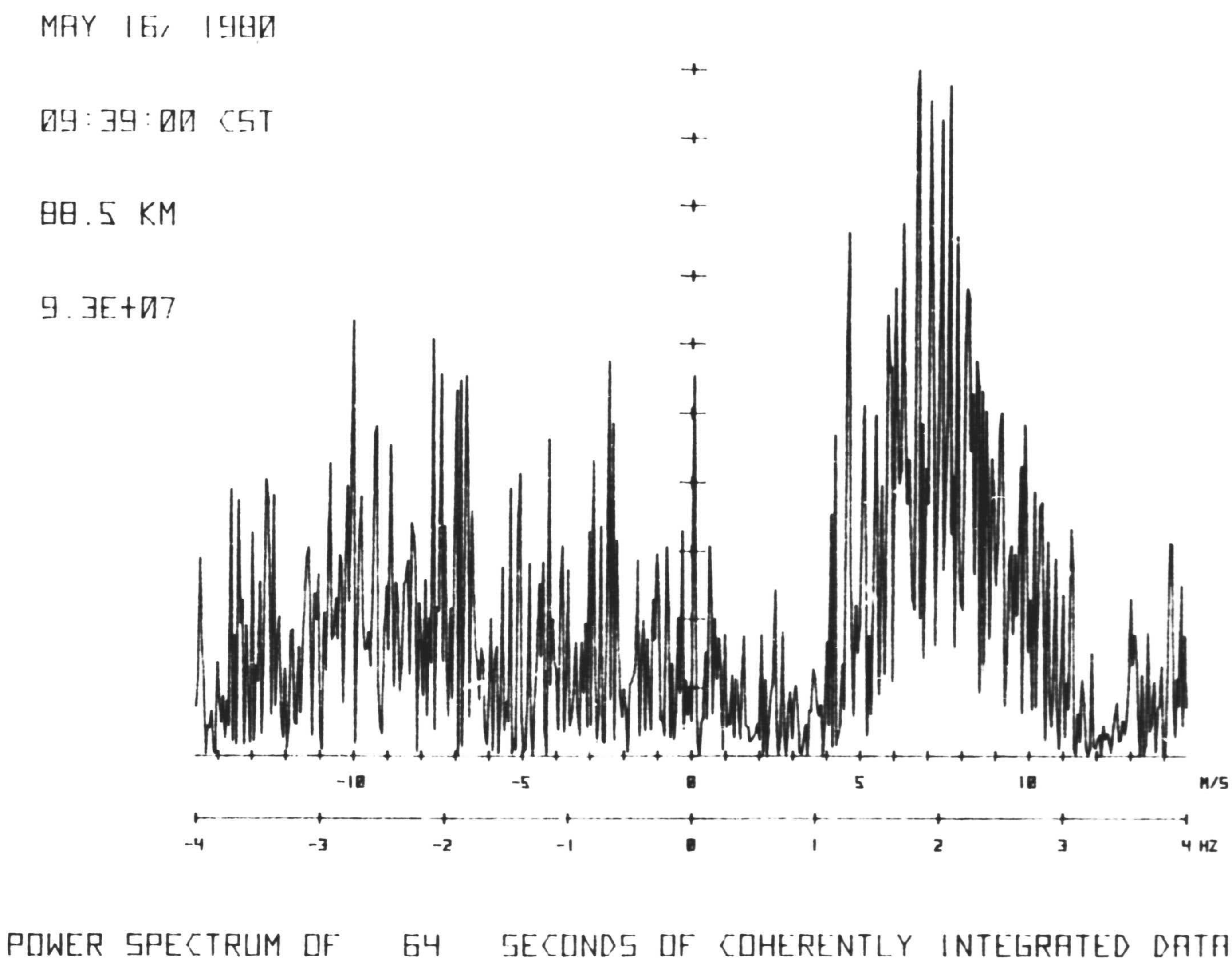


Figure 4.2 64-second power spectrum of a meteor return.

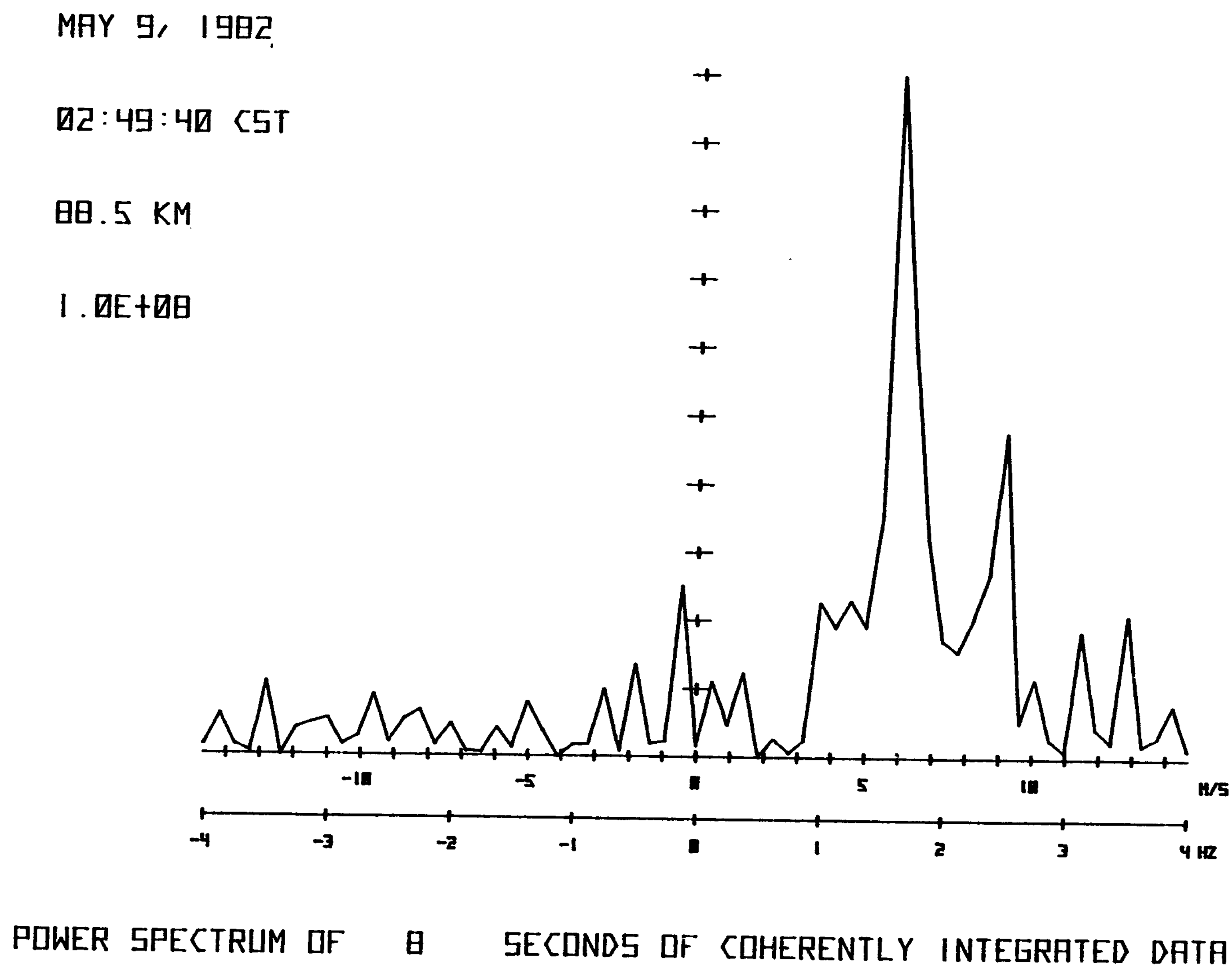


Figure 4.3 8-second power spectrum of a meteor return.

Figures 4.4. and 4.5 show two consecutive minutes of data from 72 km. These spectra are characterized by a strong return which is persistent and slowly changing in Doppler frequency. The changing velocity along the nearly vertical line-of-sight of the radar is interpreted as a gravity wave passing through the radar beam. The scaling factor in these figures is about 30 times that of the noise spectrum in Figure 4.1 indicating a large increase in power. The spectral width in Figure 4.4 corresponds to a correlation time of roughly 2 seconds.

The spectrum of data from 81 km, shown in Figure 4.6 reveals a wide range of Doppler velocities centered about a mean value of 1 m/s. The correlation time for this data is roughly 1/4 second and the power is greater than for the lower altitude spectrum shown in Figure 4.5. Variations in correlation time and power with altitude are discussed below.

Spectra which correspond to the first 8 seconds of the data used in Figures 4.5 and 4.6 are shown respectively in Figures 4.7 and 4.8. The single peak in the low altitude spectrum of Figure 4.7 is attributed to the presence of a single scattering region within the scattering volume of the radar. There may actually be more than one but in that case both are moving with virtually the same Doppler and are indistinguishable. Figure 4.8 seems to indicate an even wider spectrum with more structure than the corresponding 64-second power spectrum. This implies that some averaging gives rise to the smoother spectrum in Figure 4.6. The structure of the 8-second power spectrum is considered below.

4.5 Comparison of Spectra with Random Data

A correspondence between scattering regions within the beam and individual peaks in the power spectrum is mentioned above for the single peak illustrated by Figure 4.7. Two or perhaps three discrete peaks separated in

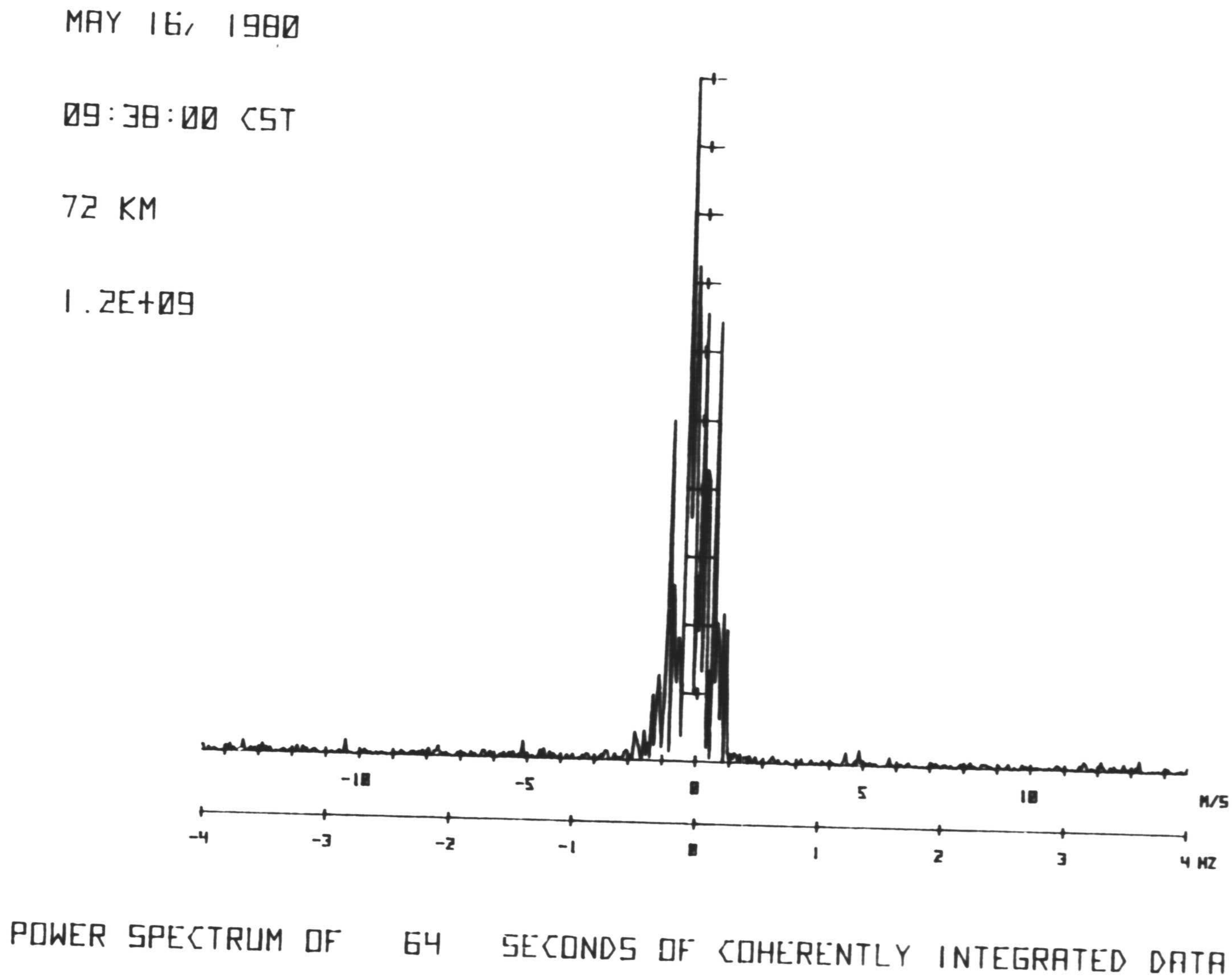


Figure 4.4 64-second spectrum from 72 km at 938 CST on May 16, 1980.

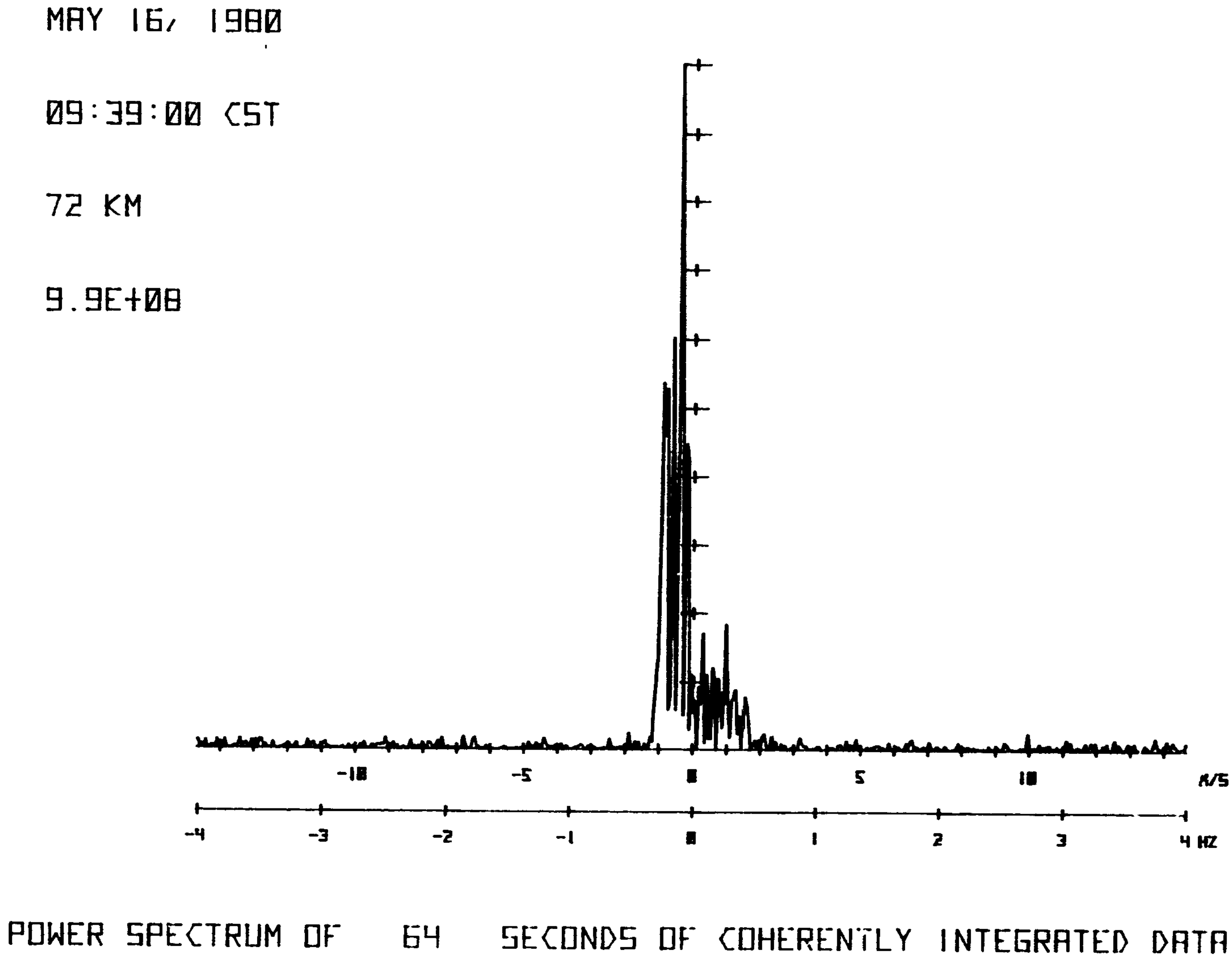


Figure 4.5 64-second spectrum from 72 km at 939 CST on May 16, 1980.

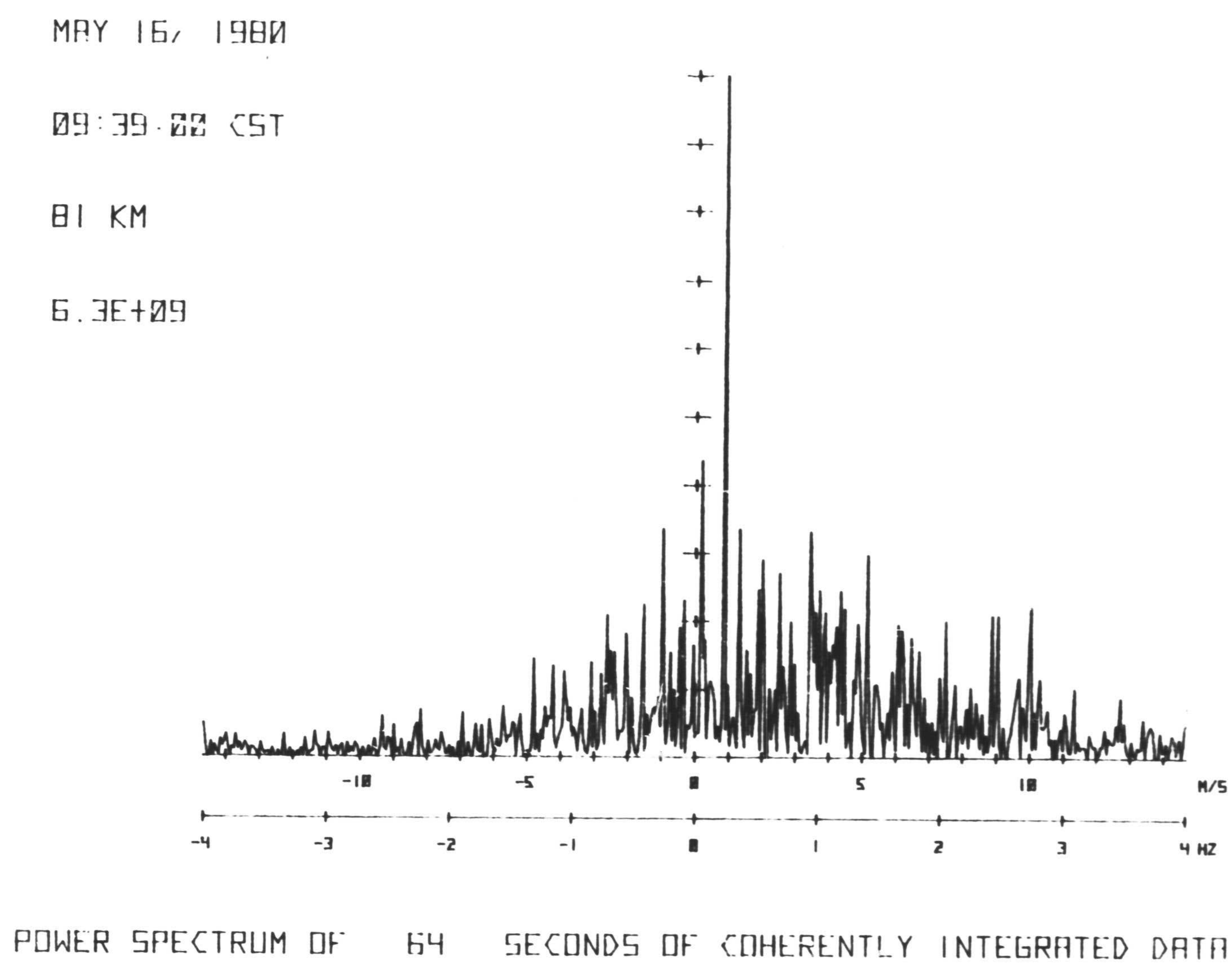


Figure 4.6 64-second spectrum from 81 km at 939 CST on May 16, 1980.

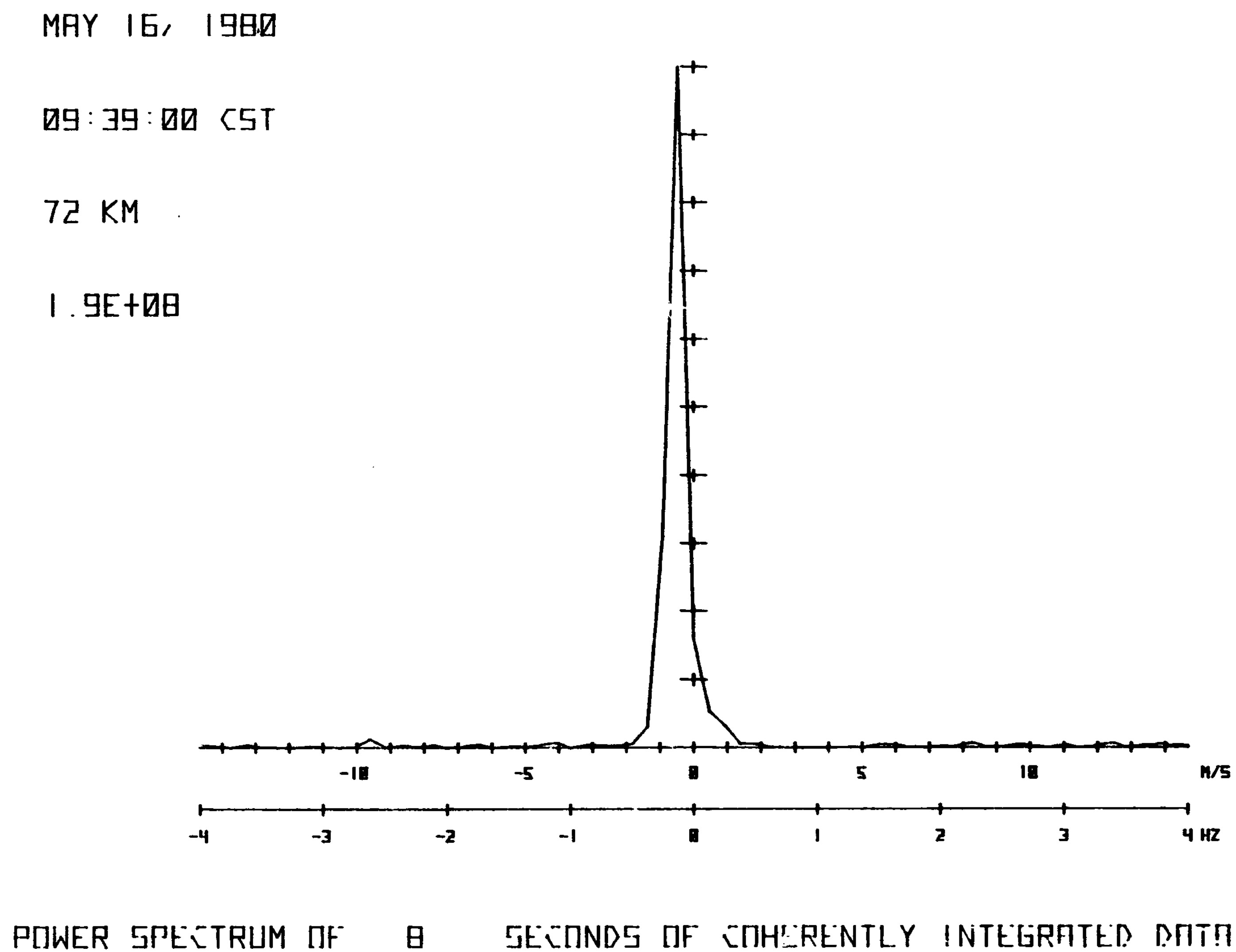


Figure 4.7 8-second spectrum from 72 km at 939 CST on May 16, 1980.

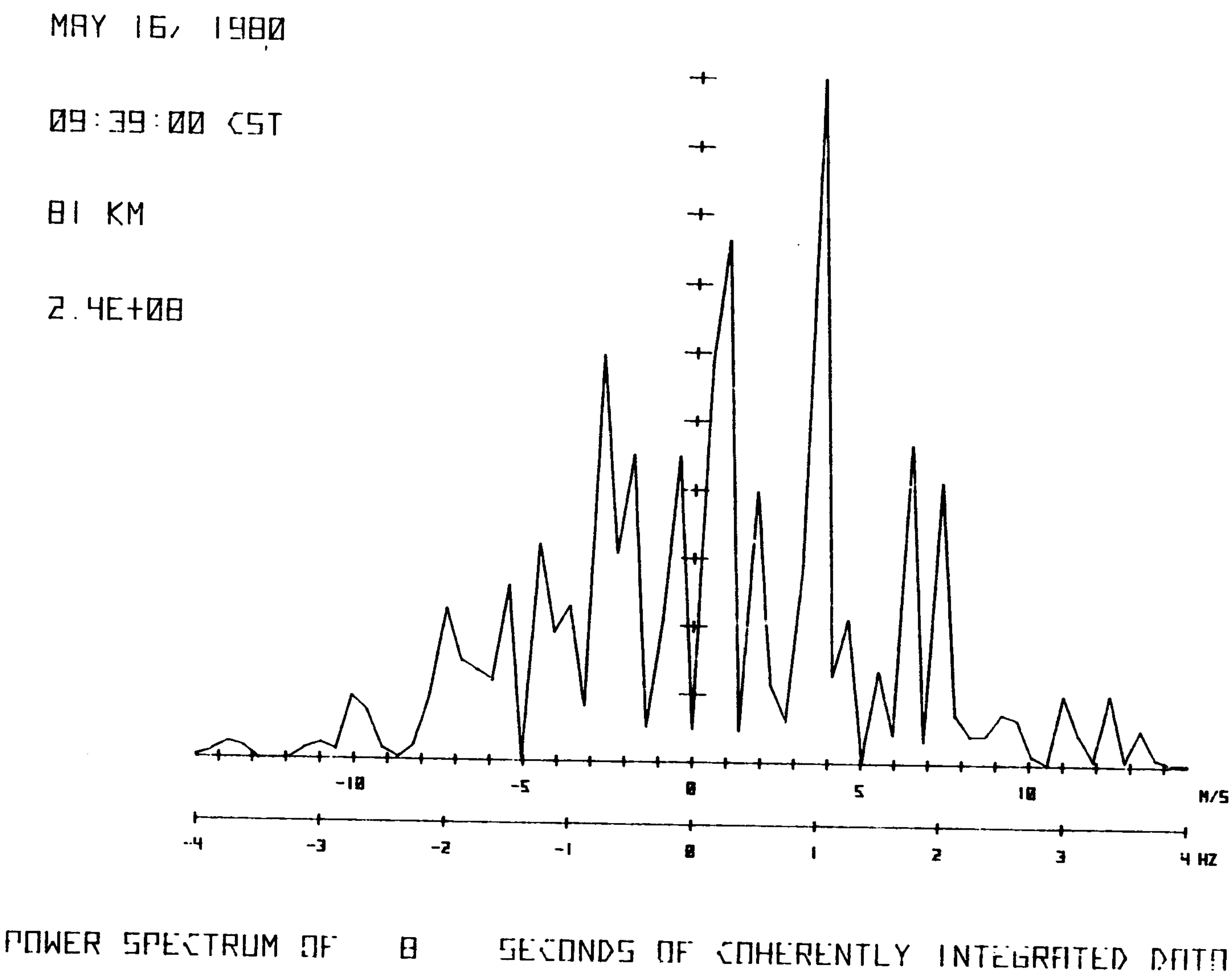


Figure 4.8 8-second spectrum from 81 km at 939 CST on May 16, 1980.

Doppler frequency can also be assigned to individual scattering regions with some difficulty. The rationale behind this correspondence is as follows: the gravity wave is effectively sampled at several physically separated points within the scattering volume by isolated regions of turbulence. The Doppler velocity at these points may differ by enough to be distinguishable so that the turbulent regions can be identified. When the power spectrum has many peaks such as in Figure 4.8 and an increased power over that of lower altitude it is reasonable to suppose that a greater percentage of the scattering volume is filled with scatterers and that subsets of these must be physically close to each other. The Doppler velocities associated with the nearly co-located scatterers should be about equal so that the returned signals will not be distinguishable and furthermore, will interact based on their relative phase. The net result is a spectrum whose peaks no longer have a correspondence to individual scatterers but rather represent an interference based on the relative motion of scatterers. Random data were generated and used to test this assertion in the manner described below.

The plots of simulated spectra shown in Figures 4.9 and 4.10 were obtained by producing a 64-second random time series, filtering with a Gaussian filter to achieve the appropriate shape, and then processing 8 seconds of the filtered sequence in the same manner as the real data. More specific details of this process are discussed in the Appendix. Comparison of the real data in Figure 4.8 to Figures 4.9 and 4.10 show that the structure of the spectrum can be due to randomness of the time series rather than to the actual presence of only a limited number of frequencies. Further analysis of spectra such as that of Figure 4.8 is therefore not possible. The analysis and results from simple spectra such as Figure 4.7 however are considered in Chapter 7.

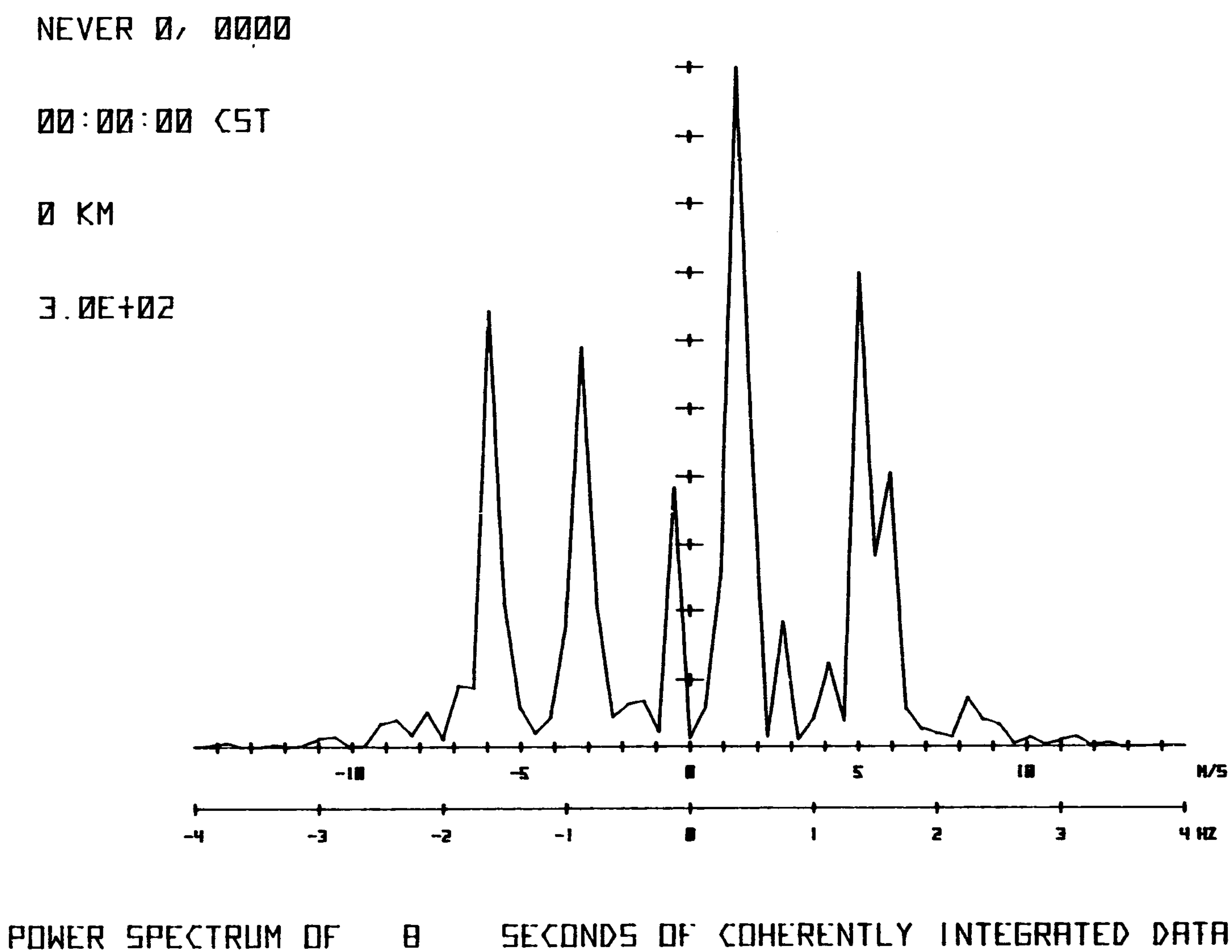


Figure 4.9 Power spectrum of a simulated, band-limited, random-phase sequence.

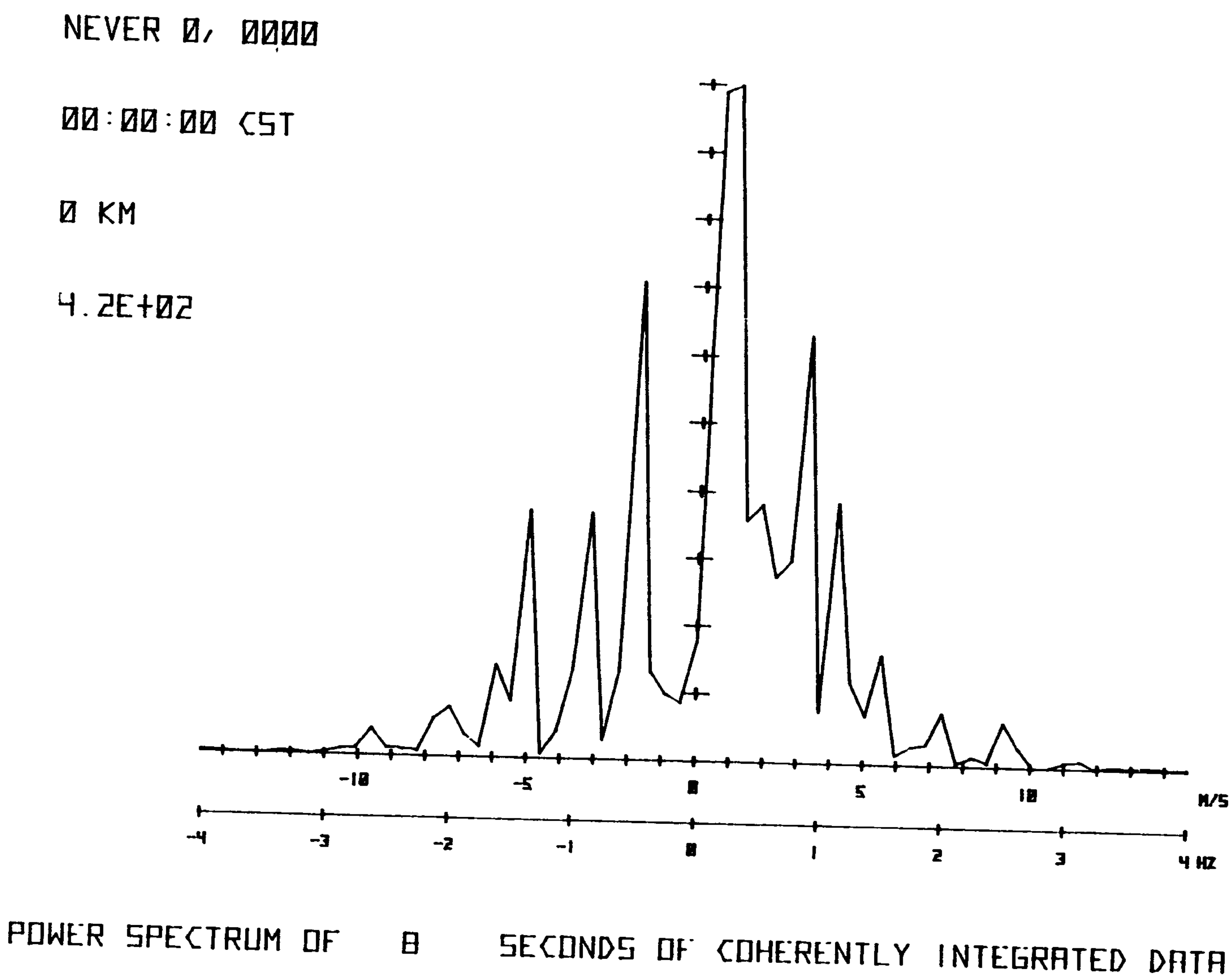


Figure 4.10 Power spectrum of simulated data.

5. DATA ANALYSIS

5.1 Introduction

The experimental technique outlined above is used in conjunction with a variety of data-processing routines to conduct the experiments for this work. One such experiment involves the accumulation of a data base from which the general characteristics of gravity-wave motion can be observed. This routine coherent-scatter experiment is reviewed in Section 5.2. A recent modification of the routine experiment has been the change to Apple II disks as the final storage media. Although this modification does not fundamentally alter the experiment, the procedures are sufficiently different to merit discussion in Section 5.3.

With the minute-by-minute data files stored on disk, additional processing of the data is possible. Routines which store and plot hourly statistics are discussed in Section 5.4. Storage of the statistics on disk allows yet another level of processing to be performed on these files to accumulate monthly, seasonal or yearly averages. Other processing of the minute-by-minute disk files includes the generation of power spectra which are also stored on disk. When velocity measurements exist for a sufficient percentage of the data file the power spectra of the velocity and for the power can be compared. The power and velocity data can also be cross-correlated. These processing routines are discussed in Section 5.5.

Finally, a data base of coherently integrated data has been collected. The processing of this data involves routines different from those used for the minute-by-minute data because the spectral analysis is performed off-line. Contour maps of power as a function of time and Doppler frequency are used to study the motion of scattering regions. The routines for the cal-

culation, transfer, and plotting of the spectra and the generation of the contour maps are described in Section 5.6.

5.2 Routine Urbana Coherent-Scatter Experiment

The coherent-scatter experiment operated at Urbana since 1978 [Miller et al., 1978] is described in detail by Gibbs and Bowhill [1979]. Recent hardware modifications such as improved phase detectors and a lower-noise receiver front end have enhanced performance but have not affected the signal processing techniques and therefore are not discussed here. A detailed description of the present hardware is given by Herrington [1983]. The purpose of this section is to review the basic operating parameters of the experiment.

The Urbana radar is operated at a pulse-repetition frequency of 400 Hz with 20-μs pulses of roughly 1 MW power. The radar is monostatic with a 3° beam width pointing 1.5° off the vertical at 36° south of east. The line-of-sight velocity therefore includes a horizontal component in the specified azimuthal direction. For each radar pulse 20 samples are collected at a 100 KHz rate from one of the two phase detector outputs. The 20 samples correspond to altitudes every 1.5 km in the 60-90 km region although the range gate can be shifted up or down. Note that the 1.5 km sampling interval represents oversampling in altitude for a 20 μs pulse. Both phase-detector channels must be sampled to obtain a complete estimate of phase. The effective sampling rate is then 200 complex samples per second per altitude.

Data are coherently integrated for 1/8 second (i.e. 25 pairs of radar pulses) by a Digital Equipment Corporation PDP-15 minicomputer. The PDP-15 also calculates the autocorrelation function to 12 lags in real time for each altitude. An estimate of the autocorrelation is obtained every 1/8 second and 480 of these estimates are summed and stored to disk each minute.

This process operates in real time until the available disk space, at two hours of collection per disk, is full. Data are then down-loaded from disk to DECtape to await postprocessing.

Postprocessing of coherent-scatter data has been modified recently to allow the transfer of data directly to Apple II microcomputer disks, a process which is described in detail in the following section. The calculation of power, velocity, and correlation time on a minute-by-minute basis is largely unchanged however. The signal-plus-noise power is given by lag zero of the autocovariance function. The line-of-sight velocity is calculated from a weighted average of the phase at lags one through three. Each phase is weighted by the value of the autocovariance at that lag so that when the correlation function changes slowly a better estimate of the phase is obtained using all three lags with roughly equal weighting. Conversely, if the amplitude and phase change quickly then the weighting favors the phase obtained at lag one. The calculation of the weighted average of phase proceeds until the autocovariance at a given lag falls below 10% of the lag-zero autocovariance. This value represents a noise threshold. If the autocovariance at lag one falls below the threshold the velocity is calculated from the phase at lag one only and a large fixed value is added as a flag to later processing programs that the particular value in question was calculated from a noisy spectrum. Finally, the width of the autocorrelation function is calculated by integrating over all lags and then dividing by the autocovariance at lag one.

The minute-by-minute data are plotted on a Hewlett-Packard 9830 desk-top computer with pen plotter. Plots of power and velocity are made from the data in a manner similar to that given by Gibbs and Bowhill [1979]. The source of the data is now Apple II disk, however, and the routines have been

improved to increase speed. The correlation data is plotted using a modification of the velocity plot routine with different scaling factors. Hourly statistics are calculated and plotted in a manner described below. All of the plots mentioned here are then added to the data base of several hundred hours of data from the mesosphere.

5.3 Data Transfer to Apple II Disk

The primary reason for the transfer of the coherent-scatter data base to Apple II disk is accessibility. The minute-by-minute power, velocity and correlation data were originally calculated on the PDP-15 and transferred to the HP-9830 by paper tape. The data were read into the HP-9830, stored on cassette, and then plotted. The limitations of the HP-9830 and of cassette storage prevented manipulation of large amounts of data to obtain monthly or yearly averages for example. With the data stored on floppy disk access is faster and large amounts of data can be quickly transferred to a hard disk for analysis routines which must search for specific files. Furthermore the Apple II can process data much faster than the HP-9830 so that plotting routines which took 30 minutes on the HP-9830 take only 3 minutes with the HP-9830 as an intelligent plotter peripheral for the Apple II. Finally the direct transfer of data from the PDP-15 to the Apple II floppy disk is faster and more reliable than the paper tape-cassette combination. The ability to collect six hours of data, process and plot it within a week represents a significant result of this work in and of itself.

The transfer of the data base from cassette to floppy disk involved three major tasks. First, the data already on cassette had to be transferred to disk. Second, the Apple II had to be able to send the data back to the HP-9830 for plotting so that no penalty would be incurred by making the switch to disk. And third, a method had to be devised to transfer data

from the PDP-15 directly to floppy disk to avoid paper tape and cassette. Therefore, the data link between the HP-9830 and the Apple II must be bi-directional while the link from the PDP-15 to the Apple II need only be one way. The specific details of the hardware and software for the transfer are given by Roth [1982] and summarized below.

The interface between the HP-9830 and the Apple II computer is a bi-directional 8-bit parallel link with hardware handshake. Speed of transfer across the link is largely limited by the speed of the HP-9830. Variables are transferred as ASCII strings rather than on a byte-by-byte basis because the latter technique would require disassembly and reassembly of the variables with BASIC statements on the HP-9830. A string of ASCII characters can be read and assigned to a variable in a single BASIC statement however. The ease and relative speed of using a single statement counter the ineffeciency of the ASCII representation of a number. The Apple II port can be controlled by machine language, BASIC, or Pascal.

The coherent-scatter data files are stored on floppy disk using Apple DOS 3.3. This operating system was chosen because of its compatibility with the BASIC and Forth languages. The transfer and plotting programs for this work are written in BASIC and compiled wherever possible to increase speed. In some cases short machine-language routines are used to handle the I/O port. Power, velocity, and correlation data files are all transferred from the HP-9830 to the Apple II using the same set of programs. The files for a given data are stored on one disk which is then write-protected. If the header information such as date, time, altitude, etc. is incorrect on a disk file another program can be used to fix the problem without requiring that the data be transferred again.

The transfer of data between the PDP-15 and the Apple II does not

involve the addition or modification of hardware to the PDP-15 but rather to a terminal attached to the PDP-15. An optical isolator is installed in a 9600-baud current loop to a terminal. An Apple II serial card with some additional hardware observes the signals on the isolator thereby giving the Apple II access to all characters sent to the terminal. Note that the Apple II has no access to the line from the terminal to the PDP-15 so that no hardware handshaking occurs between the two computers. The technique described here was chosen for two reasons. First, no additional hardware is required for the PDP-15 and only a small interface at the terminal is required. Furthermore the Apple II does not require the more complex software required to implement a complete terminal. In exchange for this simplicity the operator must act as the Apple II-to-PDP-15 handshake to indicate that the Apple II is ready to receive more data.

The transfer of power, velocity and correlation data is accomplished by operating the PDP-15 with the statements which previously punched paper tape replaced by statements which print the data. The format of a line on the terminal is modified, however, so that the Apple II serial card has time to synchronize with the data stream for each line before data characters are send by the PDP-15. Specifically, the Apple II expects to see a number of asterisks followed by 48 valid data characters. After the last character the terminal line is ignored and the characters are saved as a string. After all the strings have been sent by the PDP-15 the Apple II parses the strings into data values, obtains the header information from the operator and then stores the data onto disk. The date for the data stored on disk is indicated by a dummy file name placed on the disk during initialization of the disk. The transfer program checks for the date file in the directory in order to minimize the possibility of storing d ta under the wrong date. A

file verify program also exists so that the operator can verify a successful transfer and storage operation.

Power, velocity and correlation data are plotted using the data which is stored on the Apple II floppy disks. The Apple II disk, interface and software essentially perform the function originally performed by the cassette. The statements in the HP-9830 which read the cassette are replaced by a routine to read the ASCII format variables from the interface. For these plots the majority of time is taken by the transfer of the data and the plotter itself rather than in calculation of plot parameters. Performing the calculations on the Apple II would have increased the programming required without substantially improving the speed. The plotting programs therefore run at roughly the same speed but the overall time from postprocessing to finished plot is greatly reduced by the direct-to-disk transfer.

5.4 Calculation of Hourly Statistics

The one-minute time resolution of the Urbana radar is important to the observation of gravity waves. Longer-term variations are more easily observed in plots of hourly statistics, however. The 50% and 90% power values, the apparent horizontal velocity toward the northwest, and the standard deviation of the line-of-sight velocity are therefore calculated for each hour of data. These data were originally plotted from the cassette data and printed but not saved on cassette. Any further averaging such as in Royrvik et al. [1982] had to be carried out by hand. With the minute-by-minute source data stored on disk, the plots are now made from the disk and a second data disk is created which contains the hourly statistics discussed here and the spectra discussed below. Monthly and seasonal averages can then be calculated based on the second set of data disks.

Examples of calculations based on a month's data are shown in Chapter 6. In this section the calculation of the power statistics and the velocity statistics are discussed in detail. A brief description of the programs for plotting and disk storage is also included.

The calculation of hourly statistics for power is very straightforward. For each altitude and each hour of data in the two-hour data files the power values are sorted and the 90% and 50% values are determined. A minimum value is also determined. The minimum over all altitudes for a given hour is used as a base value and the other values are given in dB above that minimum. An example of the plot is given in Figure 5.1. The bulges above the minimum represent regions where scattering was present for at least 50% of the time, i.e., they indicate scattering layers. The difference between the 50% and 90% curves indicates the variability of the power. A large difference between the two values can also indicate a scattering region which appears only at the beginning or end of the one hour interval. The apparent motion of a scattering region with altitude can be observed with a series of plots such as Figure 5.1.

The velocity statistics plot, an example of which is shown in Figure 5.2, is somewhat more involved. Recall that the line-of-sight of the Urbana radar is slightly off the vertical in the southeast direction. A horizontal velocity component in this direction will therefore contribute to the line-of-sight velocity. The mean value of velocity observed over an hour is attributed to this horizontal component: an interpretation of data such as that of the first hour in Figure 5.2 as due to a vertical velocity requires the atmosphere to move toward an altitude of roughly 79.5 km from above and below for an hour. In addition to horizontal velocity the standard deviation and half-correlation time of the line-of-sight velocity are also calculated.

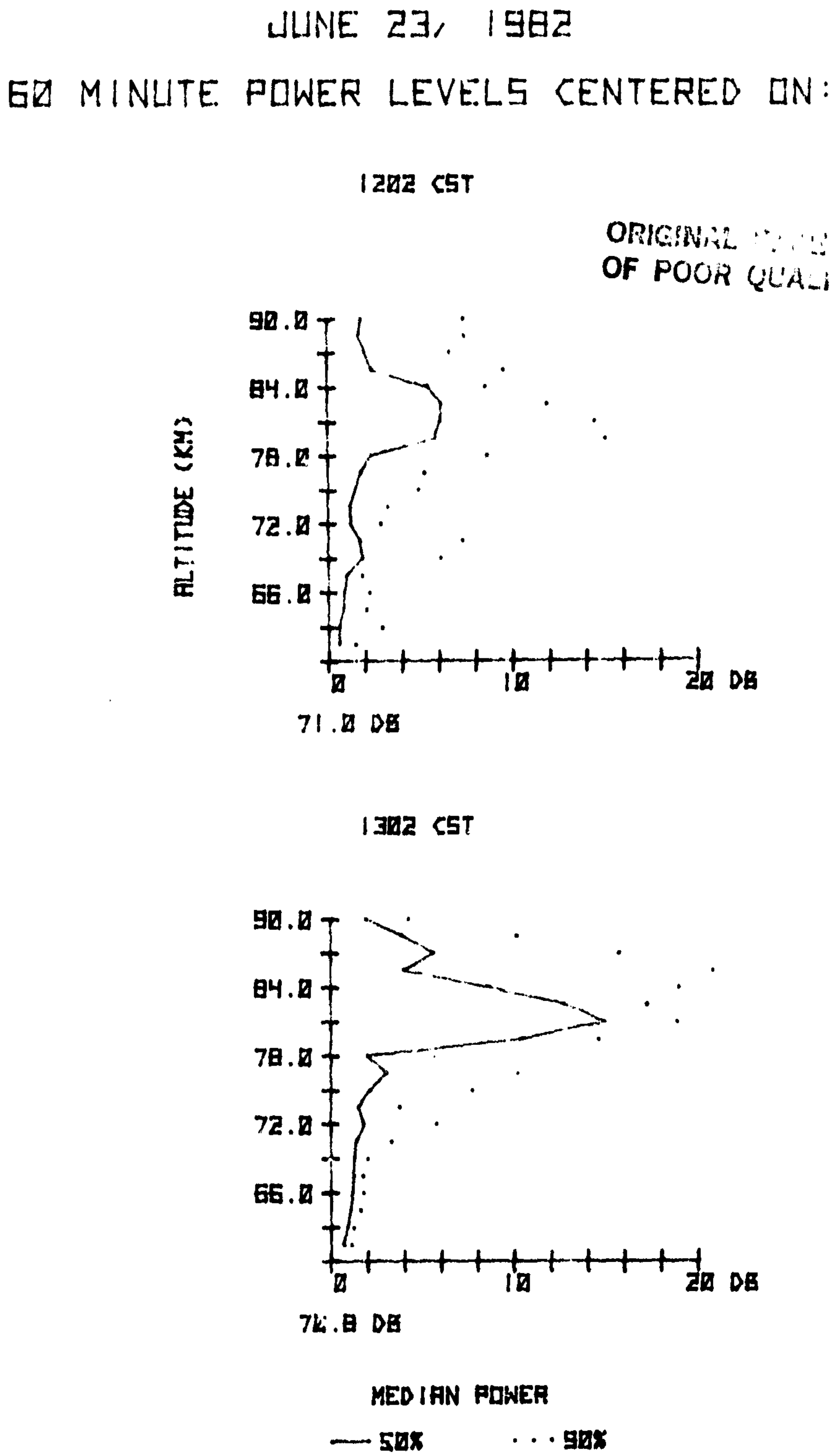


Figure 5.1 Hourly power statistics for June 23, 1982.

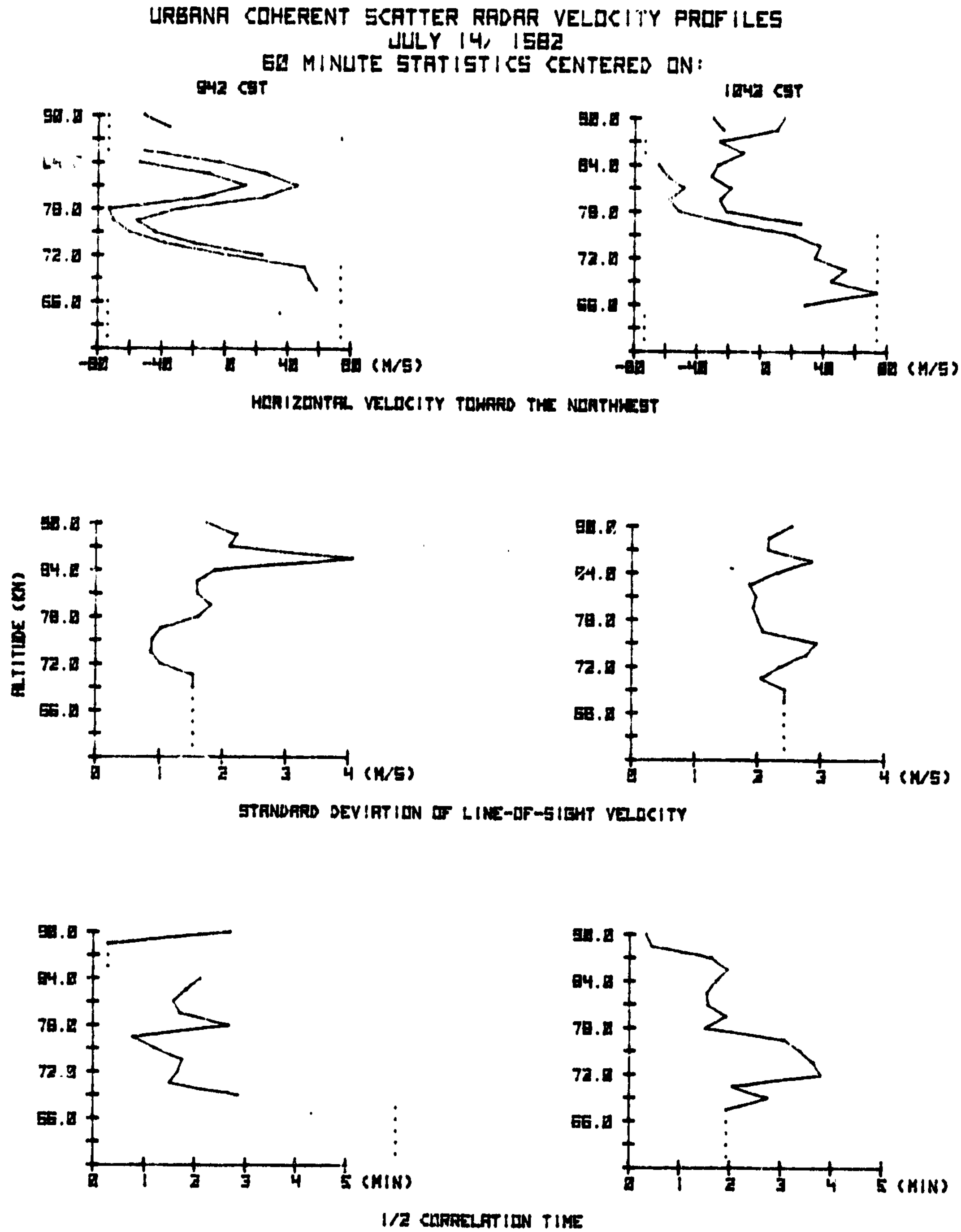


Figure 5.2 Hourly velocity statistics for July 14, 1982.

The calculation of velocity statistics proceeds in the following manner. A standard deviation is calculated from the number of points with a measurable velocity, the sum of these values, and the sum of their squares (recall that velocity data is not obtained at all times). The autocovariance of the data is used to determine the half-correlation time. The half-correlation time determines what fraction of the measured velocities represent independent sample points. If n is the number of independent points then the standard deviation calculated from the measurable velocities is multiplied by the factor $[n/(n-1)]^{1/2}$ to obtain the standard deviation of the line-of-sight velocity which is shown in Figure 5.2. The standard deviation of the mean is the line-of-sight standard deviation divided by n and multiplied by a scaling factor $1/\sin(1.5°)$ based on the antenna angle off the vertical. The horizontal velocity is the mean velocity plus and minus a standard deviation are the two curves shown in the uppermost graphs of Figure 5.2.

The programs for plotting the hourly statistics and those for storage of the plotted data are separate. Again, the desire to produce the necessary software as quickly as possible prompted this decision. The calculation routines from the HP-9830 programs were moved to the Apple II to increase speed, but no attempt was made to optimize their performance. The plotter-control statements of the cassette based versions and the statements necessary to perform the transfer from the Apple II form the new HP-9830 hourly statistics plotting programs. The plotting time is thereby reduced from roughly 30 minutes to about 3 minutes, a large part of which represents time required to draw axes and labels. Improvements in the Apple II program speed would therefore produce only marginal improvements in the overall plotting time.

The storage of the hourly statistics to floppy disk involves a relatively fast device and also requires the processing of a large amount of data. The storage program was therefore written as a bulk processing program designed to calculate, if desired, the velocity and power statistics for all the files on the data disk and to write the result files to a separate disk. Typically the program requires about 25 minutes to process a single data disk and create a second disk. Initially the entire data base had to be processed so that optimization of the storage program was important. Although the calculation routines are improved versions of those in the plotting programs the greatest savings comes from reducing the number of disk accesses. When the source and destination disk volumes are located on a hard disk instead of a floppy disk the programs run even faster. Data can be transferred between hard disk and floppy disk on a second computer, which shares the hard disk so that an assembly-line procedure can be used to process the data base. This procedure is also used for the calculation of the power spectra of the minute-by-minute data and is discussed below.

5.5 Additional Processing of Minute-by-Minute Data

A program has been developed to determine the cross-correlation of the power and velocity minute-by-minute data. The purpose of this experiment is to determine if quasi-sinusoidal variations in the power are related to the period of the gravity waves observed in the velocity data. This correlation might be observed if the turbulence is generated by the gravity wave growth with altitude as described by Hodges [1967] and discussed above. The correlation program allows the calculation of the cross correlation of two arbitrary sections from any of the power, velocity or correlation minute-by-minute data files. Various options include the use of different windows to generate the sections of data and the ability to perform the calculations

with or without removal of the mean. Finally, if the letter-quality printer is available the normalized cross correlation can be printed and a rudimentary plot made on the printer. Further discussion of the cross correlation of power and velocity data is found in the following chapter.

Another avenue for comparison of power and velocity data is through the examination of the power spectra of the data by a fast Fourier transform (FFT). The half-correlation time of the velocity contained in the hourly statistics gives some measure of the spectrum of the velocity data but no similar parameter exists for the power. The power spectrum at each altitude for all types of minute-by-minute data are therefore calculated and stored on the same disk as the hourly statistics. The mean value is subtracted and zeros are added to make the FFT always of length 128. The power spectrum is given by the magnitude squared of the FFT. This program is also designed to work with large amounts of data. The disk handling and operator interface sections of the power spectrum program are virtually identical with those of the hourly statistics storage program discussed above. The FFT is an in-place, complex, power-of-two algorithm with the coefficients calculated external to the FFT. Furthermore, the first two stages of the FFT are calculated without multiplication at the expense of separate code. Because 20 FFTs are calculated per file with typically 9 files or more per disk the savings in time becomes substantial. Each FFT requires approximately 6 seconds to calculate. Generally a single disk requires about 45 minutes to process and store to the output disk.

Both the power spectra program and the hourly statistics program can be used for single disks of data as they are collected. Initially the entire data base had to be processed through these two programs, however. An EXECute file which contains all the responses to the input prompts from

either the Apple II operating system or the user programs is created. When the disk operating system is told to EXEC that file the data in the file is used to control the Apple II. No operator intervention is required until the end of the EXEC file is reached. Programs can therefore be sequentially and repeatedly executed if the required data are available on hard disk.

The EXEC file is used in conjunction with the power spectra and hourly statistics programs to process multiple data disks. The data from several (usually 6) floppy disks are transferred to a block of volumes on the hard disk. An identical number of destination volumes are cleared by transferring onto hard disk from a special floppy disk which contains only the dummy data file as discussed above. This avoids the time-consuming operation of resetting the user-access tables on the hard disk which would be required if the hard disk volumes were cleared by re-initialization. The data on each destination volume is then changed to correspond to one of the source volumes. A program which creates the appropriate EXEC file uses the volume numbers, file starting times and other input. Under control of the EXEC file the Apple II runs for many hours, typically overnight. The newly created destination disks are copied from the hard disk onto floppy disks and the procedure repeated. The programs and file structures are discussed in detail in the Appendix.

5.6 Collection and Transfer of Coherently Integrated Data

In addition to the data processing for the minute-by-minute data discussed above a second set of processing routines has been developed for the coherently integrated data. The primary purpose for the collection of coherently integrated data is detailed analysis of the spectrum of the returned signal. The first stage of postprocessing therefore largely consists of programs which calculate power spectra and store the data on

Apple II floppy disks. Subsequent analysis proceeds from these disk files. The various programs are discussed below.

The collection of coherently integrated data is largely the same as the routine coherent-scatter experiment with the exception of a different computer program for the PDP-15. The collection program puts the coherently integrated data directly on disk rather than calculating an autocorrelation. Disks are filled with forty records of 10 seconds length. As before the collection stops when all available disk space is full. The data are then dumped to DECtape.

A postprocessing program for the PDP-15 calculates the power spectra at all altitudes in increments of 30, 60, or 120 seconds for an entire disk of input data. The routine used to calculate the power spectra is the FORTRAN version of the high-speed FFT program discussed in the previous section. In order to obtain a power of two with 8 samples per second the actual data length is 32, 64, or 128 seconds. Therefore a small overlap occurs between FFTs. The power spectra are normalized to minimize the number of characters which must be sent to the Apple II and stored on disk. The transfer occurs in the same manner as described above for the routine data collection. In this case, however, each disk of data on the PDP-15 is being mapped to many files on the Apple II, one for each FFT. Furthermore the PDP-15 can calculate faster than the Apple II can write to disk so that a handshake is required to control the transfer. The operator therefore toggles a console switch on the PDP-15 to send each FFT. In turn the Apple II prompts the operator with a visual and audio indication when it is ready for more data. Transfer of a PDP-15 disk of coherently integrated data with 60-second FFT spacing requires about 45 minutes. The transfer speed can be improved by using the Apple II hard disk for transfer and a second Apple II

to copy to floppy disk in the background. The files of power spectra are plotted using the HP-9830 and techniques similar to those used for the routine data.

Analysis of a number of 64-second power spectra led to the conclusion that a power spectrum of a shorter data set would be desirable. A program has been written to calculate power spectra from the first 8 seconds of each 10-second record on the PDP-15 disk. Because of the large number of spectra involved the operator can choose individual times and altitudes to be calculated and transferred. The entry of the time and altitude data serves as the handshake to control transfer for this program. The selection process is based on a printout from the PDP-15 of power for each 10-second record. A similar program prints the power values for 1-second intervals. In general the 10-second power program is run first to examine the overall quality of the data and determine which FFTs should be calculated. If the data indicates a high degree of variability such as for a meteor echo then the 1-second power program is also run. The power spectra are then calculated and transferred.

A contour map of power vs. Doppler frequency and time is printed from the floppy disk files. The power in dB above a selected reference forms a grid with a row for each time at which the spectrum was measured and a column for each Doppler frequency bin. Contours are then hand-drawn around 10 dB power levels. A contour map is drawn for several adjacent altitudes covering the extent of the scattering layer. Individual scatterers are identified by significant power levels in adjacent Doppler frequency bins for several 10 second time periods. The data from the appropriate floppy disks are transferred to hard disk so that a large number of spectra can be read by a program which calculates the altitude of specific components.

This program must look at the value of a particular component for five altitudes which bracket the layer. The maximum of the three middle values is used as the center for a parabolic fit to the variation in power. The calculated altitude for all components within a given range of Doppler frequency bins is printed for each 10-second time period. The altitude and Doppler frequency of the individual scatterers identified previously are then plotted. The results of these calculations are shown in Chapter 7.

6. RESULTS FROM THE MINUTE-BY-MINUTE DATA

6.1 Percentage of Data with Measurable Velocities

The measurement of a Doppler velocity from the coherent-scatter signal is dependent on the signal-to-noise ratio as described above. Furthermore, the mixing-in-gradient process responsible for the scattered signal requires that the signal-to-noise ratio be dependent on the electron density which is in turn dependent on the solar flux. The amount of signal should therefore depend on the time of day.

A measurement of the percentage of time for which measurable velocities were calculated for April 1978 is shown in Table 6.1. Note the general increase from early morning toward noon followed by the decrease in the evening. There is also a broad altitude dependence in the data which is illustrated for the noon hour in Figure 6.1. The increase with altitude at lower altitudes is the result of an increasing electron density with altitude. At higher altitudes the energy at 3.7 m in the spectrum of the turbulence decreases with altitude overtaking the increase in electron density. The highest percentages, therefore, occur in the 70-75 km region near midday. These data represent one example of the type of processing possible with the data base stored on Apple II disk.

6.2 Nighttime Echoes

In general, it is not possible to measure continuous velocity data at night with the coherent-scatter experiment. Ionization from meteors or from unusual solar activity can occasionally enhance the electron density to a useable level, however. Echoes from meteors are discussed in the following section. An example of unusual solar activity is discussed below.

The occurrence of unusual solar activity during daylight observation of

Table 6.1 Percentage of measurable velocities for April 1978.

APRIL.1978
PERCENTAGE OF MEASURABLE VELOCITIES

ALTITUDE (KM)	TIME(CST) 4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
87		14	28	38	43	52	59	50	49	64	45	56	49	54	33	10	
85.5		10	24	35	38	43	54	50	59	73	54	61	47	47	28	32	
84		9	27	38	45	55	57	65	62	70	67	58	54	39	18	30	
82.5		13	34	51	57	55	53	76	62	63	71	76	63	43	16	15	
81		10	40	55	59	52	59	79	68	61	64	73	73	46	11	28	
79.5		12	36	68	66	53	68	73	77	72	60	75	79	47	14	33	
78		18	42	74	69	77	72	77	79	72	70	80	76	47	19	35	
76.5		23	32	67	73	84	79	73	89	81	68	64	68	42	12	18	
75		18	24	54	64	82	81	82	85	82	77	68	62	41	17	15	
73.5		17	14	40	63	86	94	91	88	80	77	70	66	40	21	0	
72		30	15	48	79	89	99	98	93	88	79	82	65	52	39	3	
70.5		51	36	61	86	93	99	98	92	89	80	80	70	67	54	12	
69		55	49	71	86	87	88	93	82	83	71	78	81	75	51	18	
67.5		54	61	80	80	83	84	89	67	73	68	80	76	75	45	8	
66		39	53	63	68	72	75	71	65	60	45	61	48	51	35	12	
64.5		26	40	33	52	61	61	52	49	40	26	32	25	25	31	2	
63		17	32	18	39	38	50	38	30	35	23	21	21	21	30	3	
61.5		10	14	14	28	20	15	14	14	19	11	11	10	10	18	10	
60		5	7	4	10	8	6	8	6	7	8	11	5	6	9	15	
58.5		4	7	4	10	7	8	7	4	6	5	10	6	5	5	8	

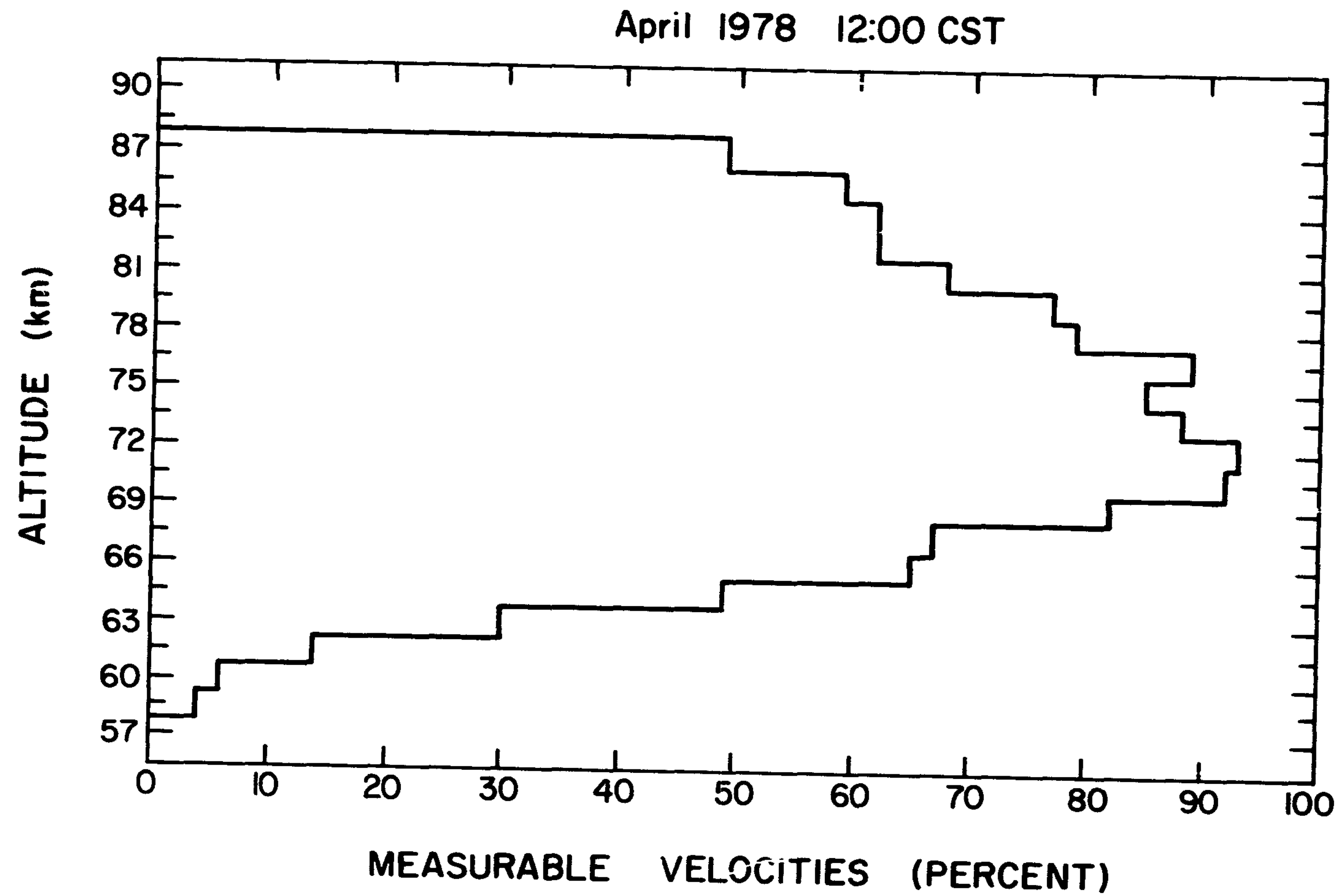


Figure 6.1 Noontime measurable velocity percentage for April 1978.

coherent scatter is shown by Miller et al. [1978]. The continuous observation of scattering at virtually all altitudes is evidence for increased electron density due to an influx of solar particles. At nighttime the photoionization process does not occur so that the effect of the unusual solar activity is less dramatic. Data collected during a geomagnetic disturbance during the period April 24 to April 26, 1982 are shown in Figures 6.2 to 6.5. In Figures 6.2 and 6.3 the scattering is sufficient to produce nearly continuous measurement of velocities for over an hour. Furthermore, the apparent motion of the scattering region is similar to that observed in daytime measurements. This suggests that the generation of turbulence is unchanged at night. Figure 6.4 shows the continuous velocity curves expected for the enhancement. Note the absence of turbulence above 87 km in this data set. Finally, in Figure 6.5 echoes are obtained on the night following the peak of the storm. The returns show less continuity which is an indication of a reduced signal-to-noise ratio. Essentially the increased electron density allows observation of turbulence which would otherwise not be seen. The result is the observation of only the strongest turbulence at night but so many turbulent regions during the day as to be beyond the range resolution of the radar.

6.3 Echoes from Meteors

Echoes from meteors are observed at Urbana whenever the coherent-scatter radar is operated. The meteors appear in the minute-by-minute data as a one minute burst of power often 10 dB or more above the noise power. The velocity data shows deviations from a smooth curve at these times because the ionization produced by the ablation of the meteor alters the relative contribution of the various turbulent cells within the scattering volume. The observed Doppler velocity, which is a composite of the Doppler

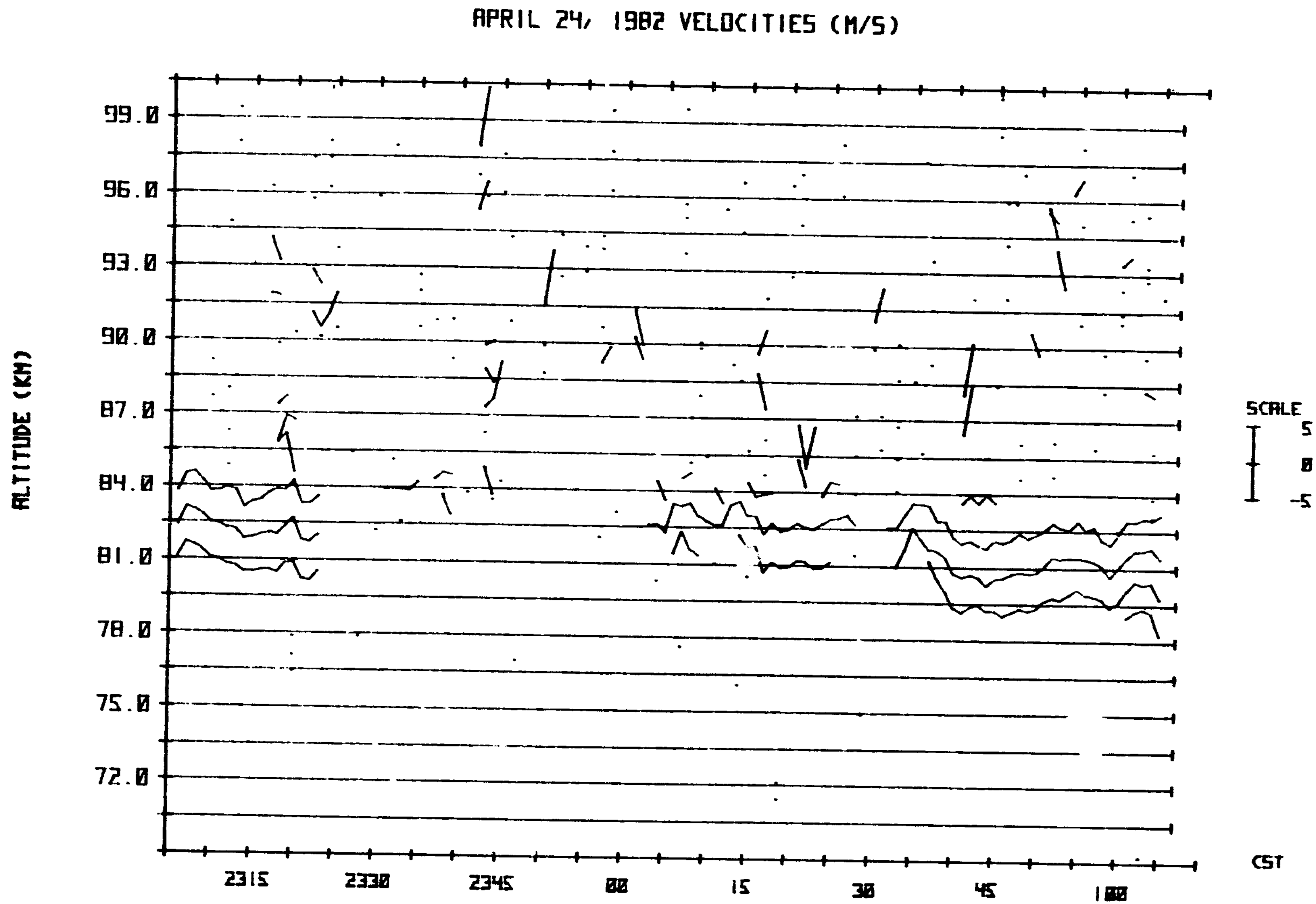


Figure 6.2 Nighttime velocity measurements on April 24, 1982.

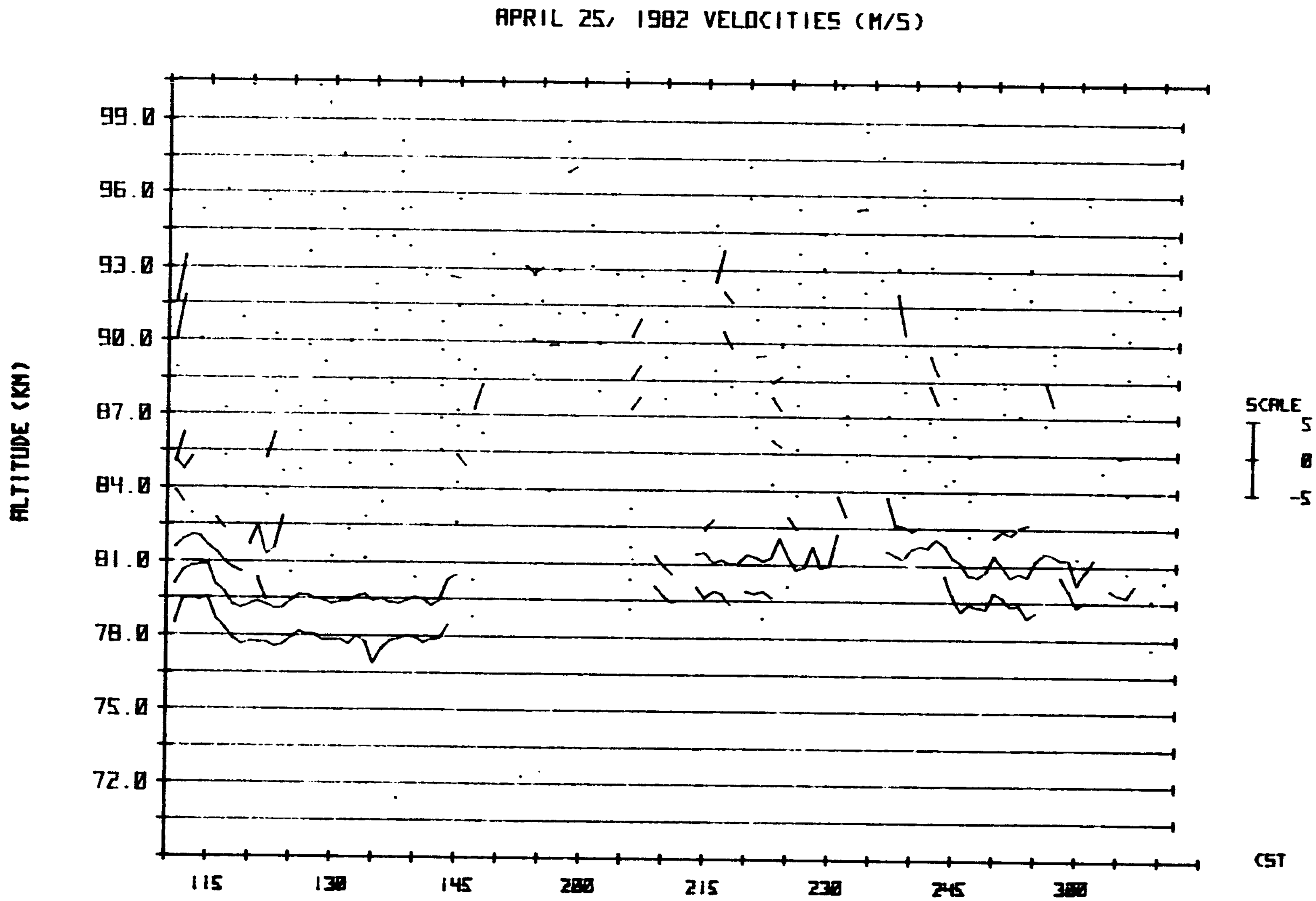


Figure 6.3 Nighttime velocity measurements on April 25, 1982.

ORIGINAL PAGE IS
OF POOR QUALITY

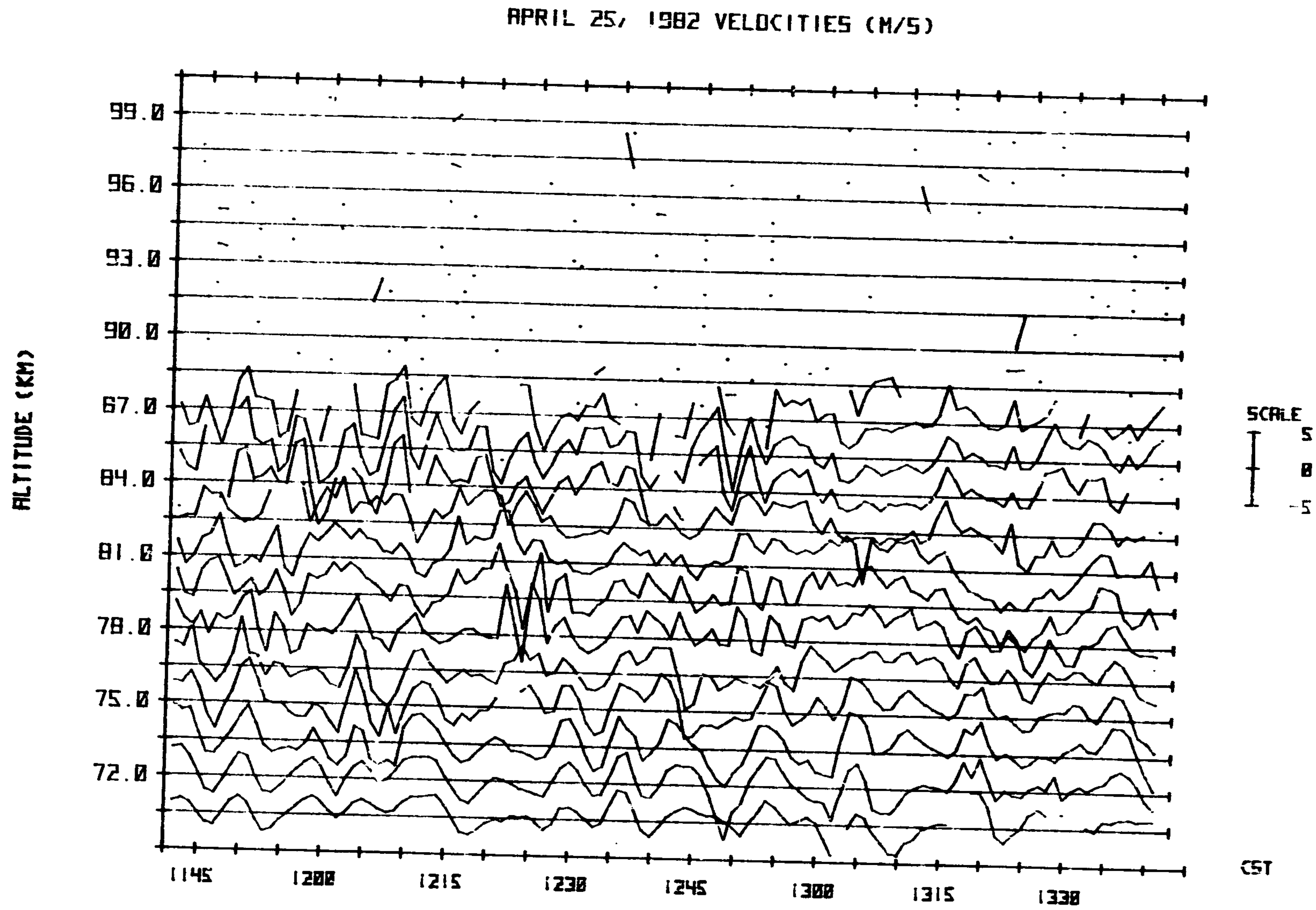


Figure 6.4 Noontime velocity measurements on April 25, 1982.

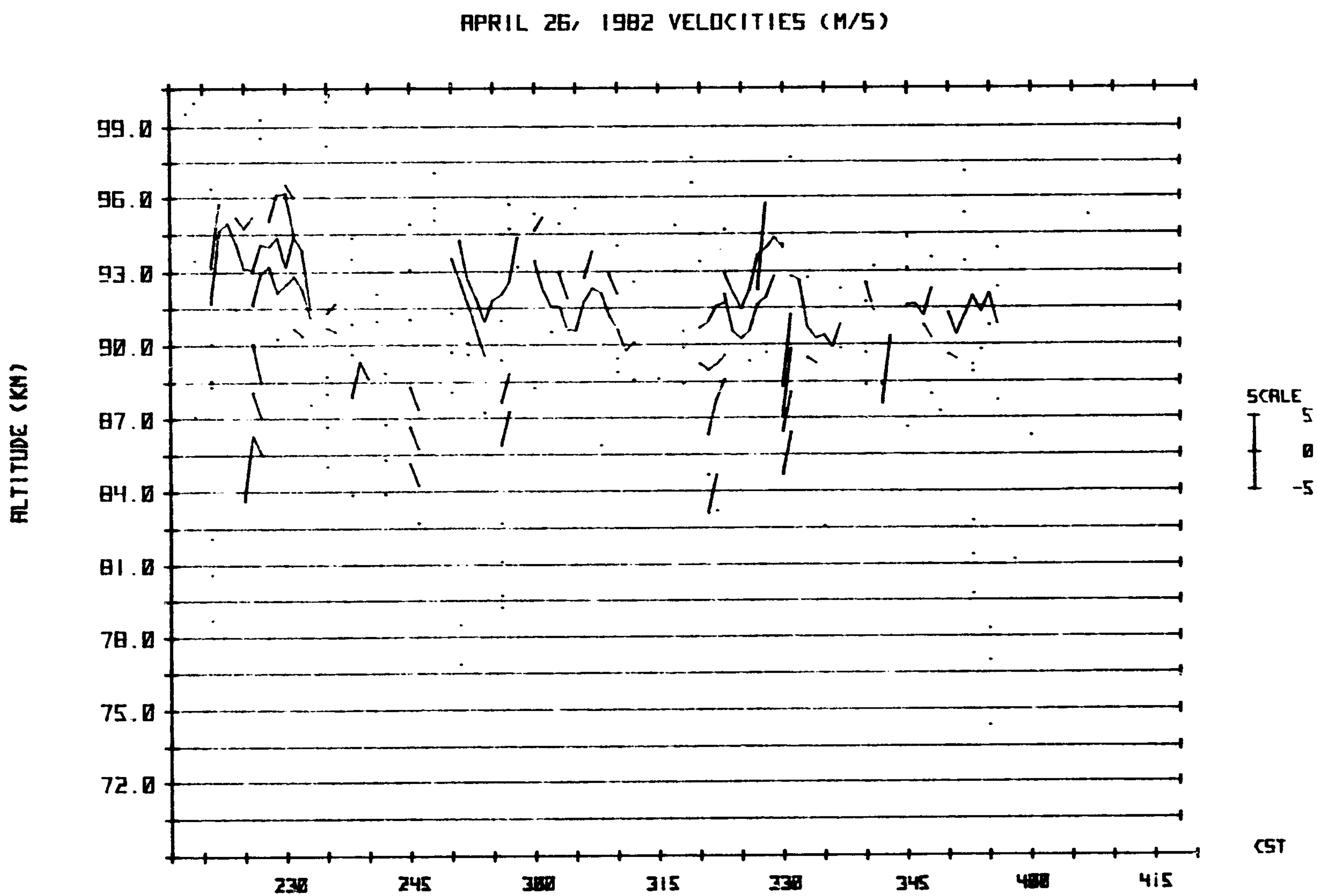


Figure 6.5 Nighttime velocity measurements on April 26, 1982.

velocity of the various scatterers is therefore altered.

At night the ionization from the meteors is usually the only source of ionization sufficient to allow coherent-scatter echoes. An example of echoes obtained during the Eta Aquarids shower is shown in Figure 6.6. On a few occasions a velocity value is obtained for consecutive minutes but generally the data are unusable for the observation of waves. A plot of power integrated for one second is shown in Figure 6.7. An increase of power of about 15 dB in the altitude range 85.5 km to 90 km near 02:49:30 CST is attributed to a meteor echo as are the smaller peaks at 90 km from 02:50:00 CST to 02:50:10 CST. Note that the echoes are typically a few seconds in duration. The long lasting echoes at 88.5 km and 87 km are likely to be the result of separate meteors at 02:49:38 and 02:49:41 CST.

A different situation is shown in Figure 6.8 for data collected during the Arietids shower which consists of particles smaller but more numerous than those in the Eta Aquarids shower. The velocity data are very ragged but gravity-wave motion is observable. The turbulent layer again descends with time similar to the data observed during daylight. Clearly if the meteors occur often enough then a nighttime coherent-scatter experiment is possible at the highest altitudes in the mesosphere.

6.4 Comparison of Power and Velocity Variations

The comparison of variations in power to those in velocity is based on the occurrence of data such as that shown in Figure 6.9. The quasi-sinusoidal variation in power from 8:37 to 9:07 CST at 75 km specifically is of interest. If a gravity wave is of sufficient amplitude then regions of shear-induced turbulence should propagate with the wave as discussed above. And because the greatest shear occurs at the zero-crossings of the wave the turbulence could be expected to occur with a spatial separation half that of

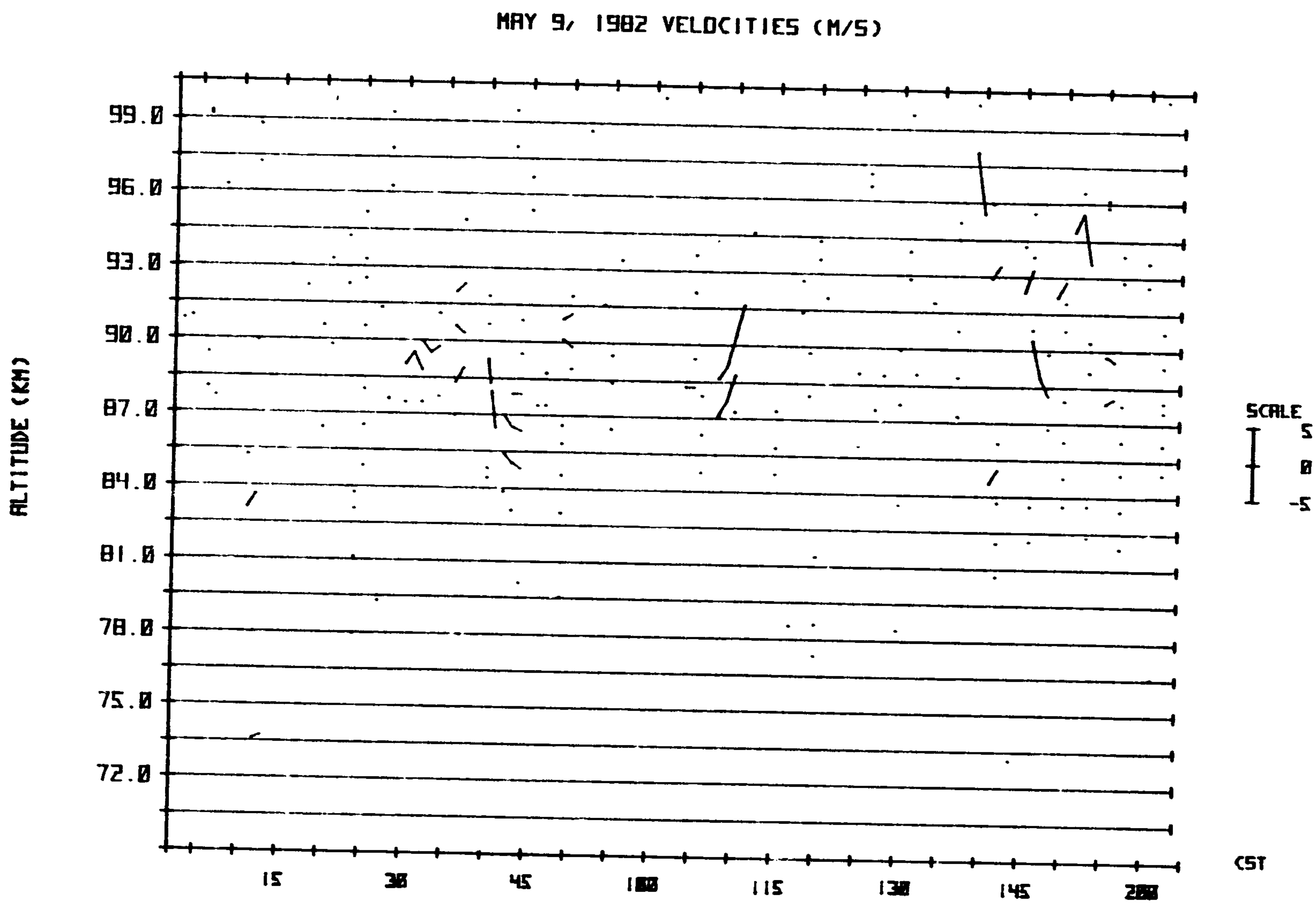


Figure 6.6 Velocity data measured during the Eta Aquarids meteor shower.

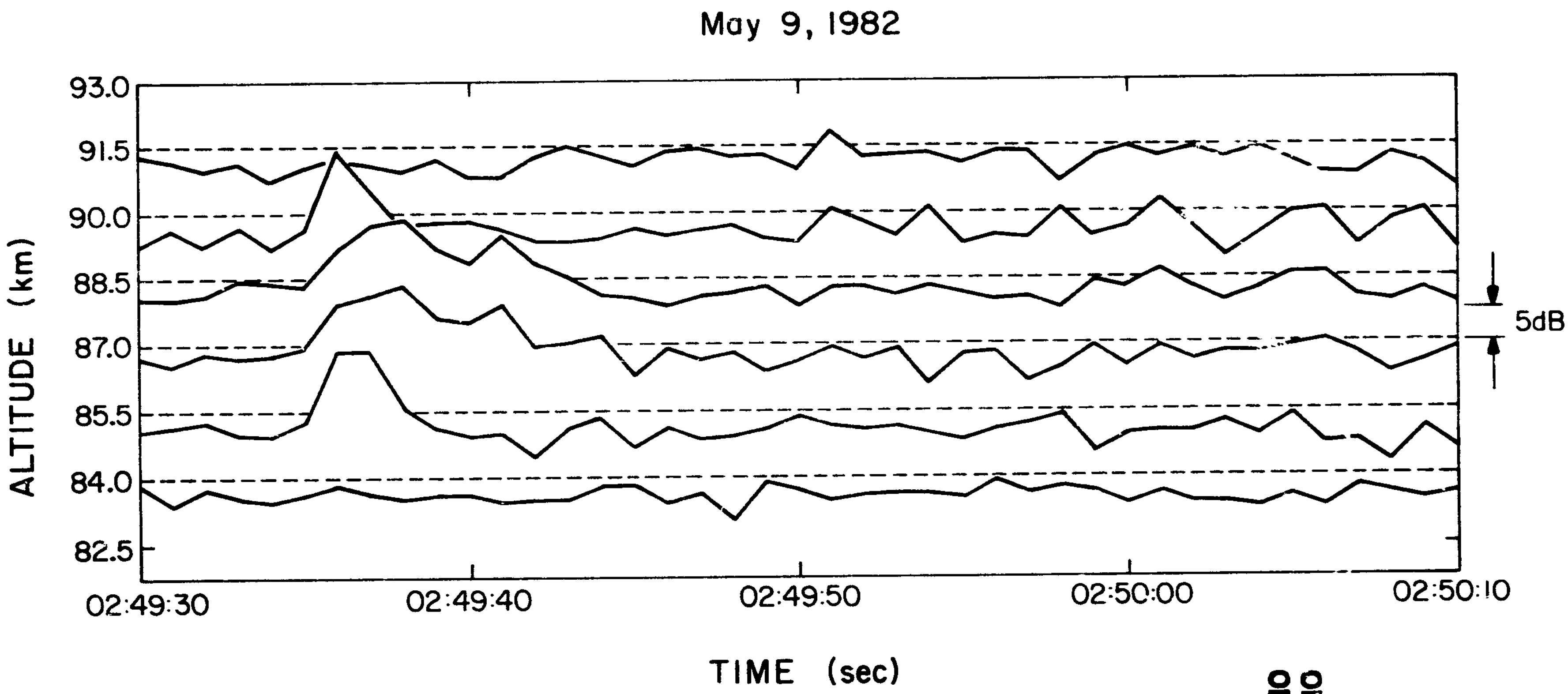


Figure 6.7 Power data from the Eta Aquarids shower.

ORIGINAL PAGE IS
OF POOR QUALITY

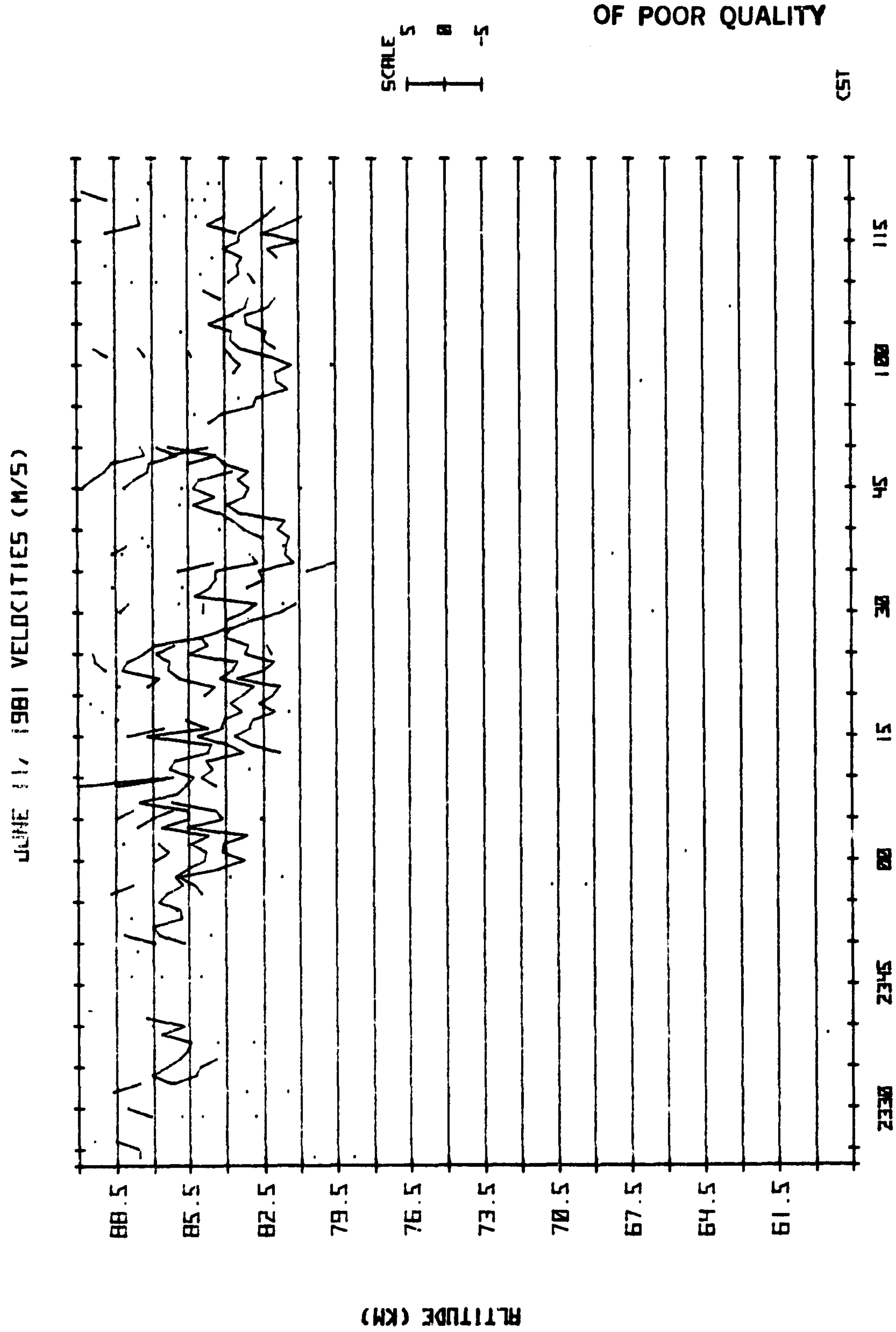


Figure 6.8 Velocity data obtained during the Arietids shower.

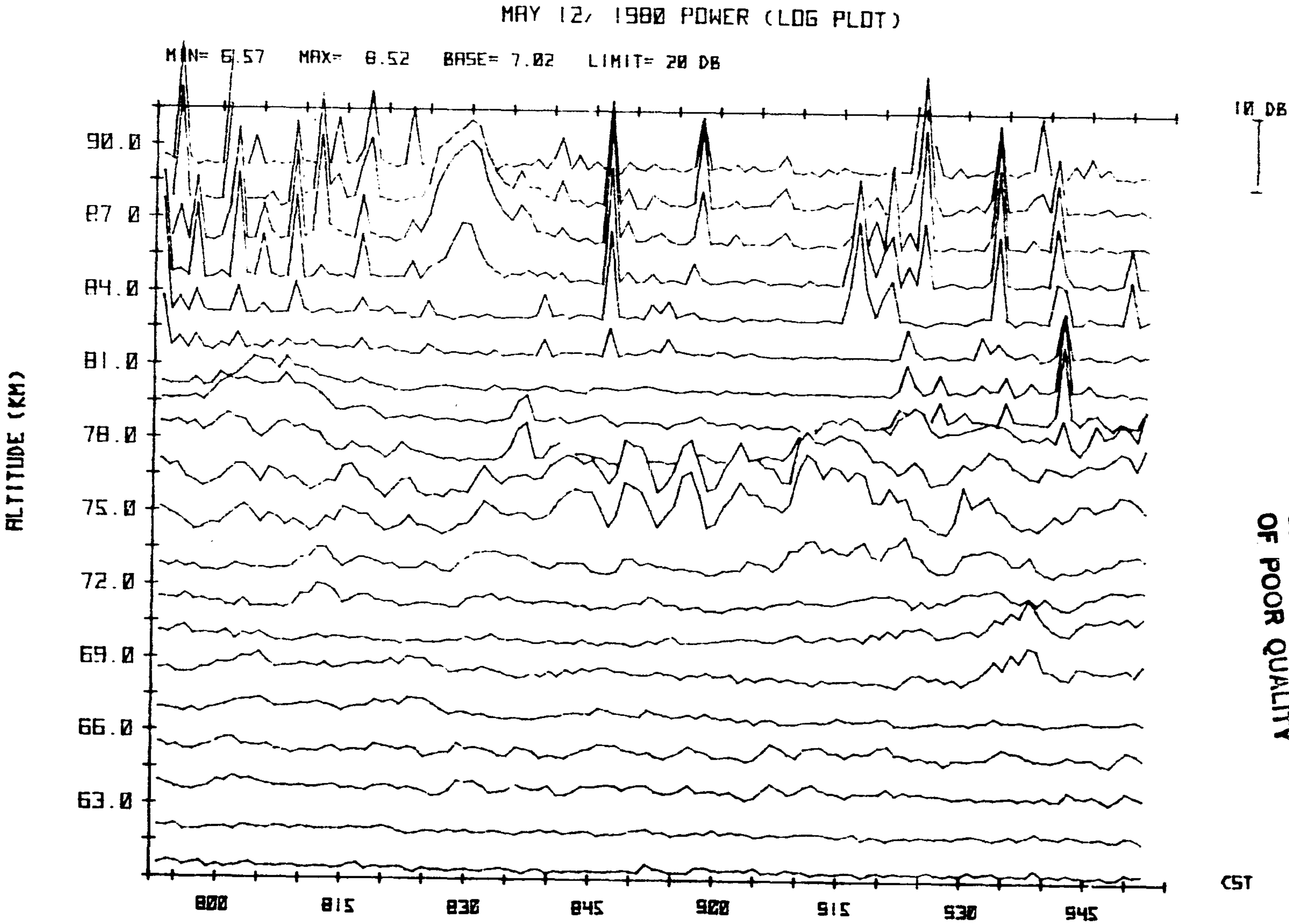


Figure 6.9 Variations in power on May 12, 1980.

the gravity wave. The variations in power observed by the radar would then have half the period of the gravity wave.

The velocity curve which corresponds to Figure 6.9 is shown in Figure 6.10. For roughly 15 minutes between 8:45 and 9:00 CST the hypothesis is satisfied at 75 km but is not satisfied elsewhere. The power spectra for the power and velocity data are shown in Figure 6.11. The amplitude of the power spectra are plotted linearly against the period in minutes and scaled so that the highest spectral peak corresponds to full scale on the plot. The spectrum of the velocity peaks at a 12.8 minute period while the spectrum of the power peaks at 6.7 minutes, nearly half of 12.8 minutes. The other peaks in the velocity spectrum do not fit this model, however. Figure 6.12 shows another example of power data with a discernible peak in its power spectrum. Again the spectrum of the velocity shows a peak at roughly twice the period of the peak in the power data.

In contrast to the data above the power data do not usually produce a distinct peak in the spectrum because the quasi-sinusoidal variations are present only for a fraction of the two-hour data set. However, the velocity data for the same time period often shows quite distinct spectral peaks. The small number of examples where the quasi-sinusoidal power variation is observed and the limited agreement with the hypothesis above imply that the generation of turbulence in layers moving with a gravity wave is not a dominant mechanism.

It may be argued that the presence of a horizontal wind shear will modify the above hypothesis. Specifically the horizontal shear will aid the wave-induced shear at one zero-crossing and counteract at the other zero-crossing. In this case a one-to-one correspondence between the velocity and power variations would be expected. There is some evidence for this in the

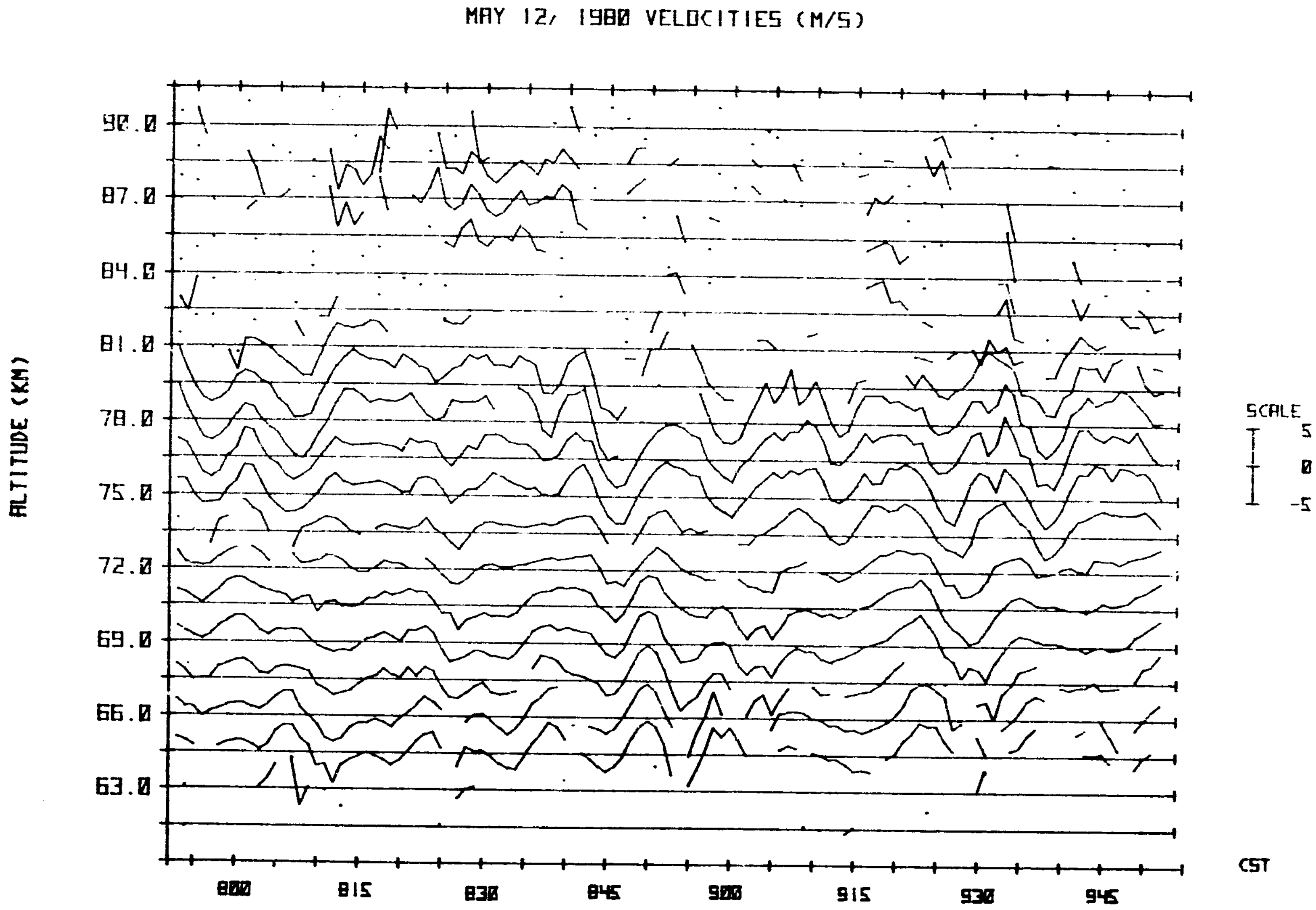


Figure 6.10 Velocity data from May 12, 1980.

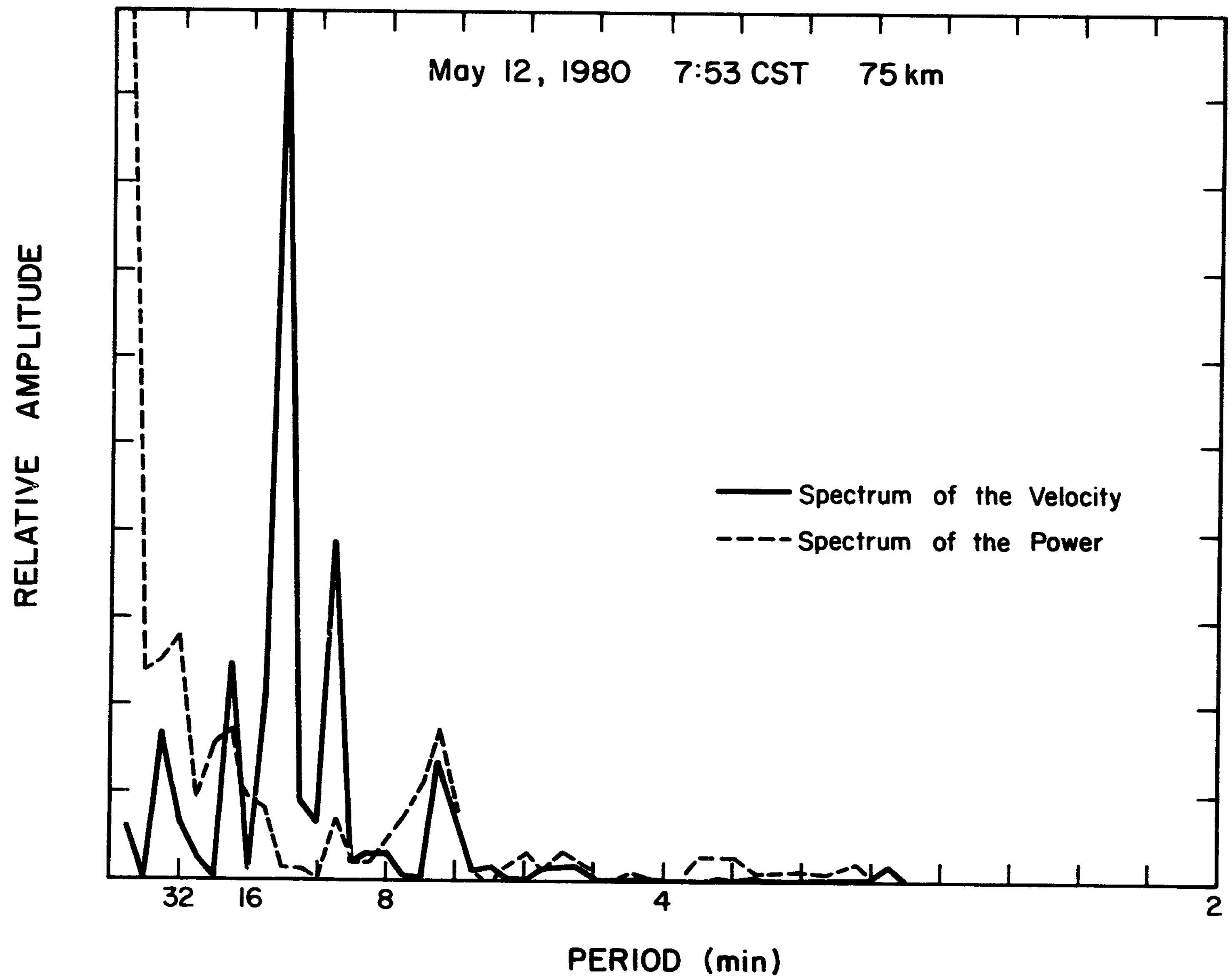


Figure 6.11 Power spectra for the power and velocity variations on May 12, 1980.

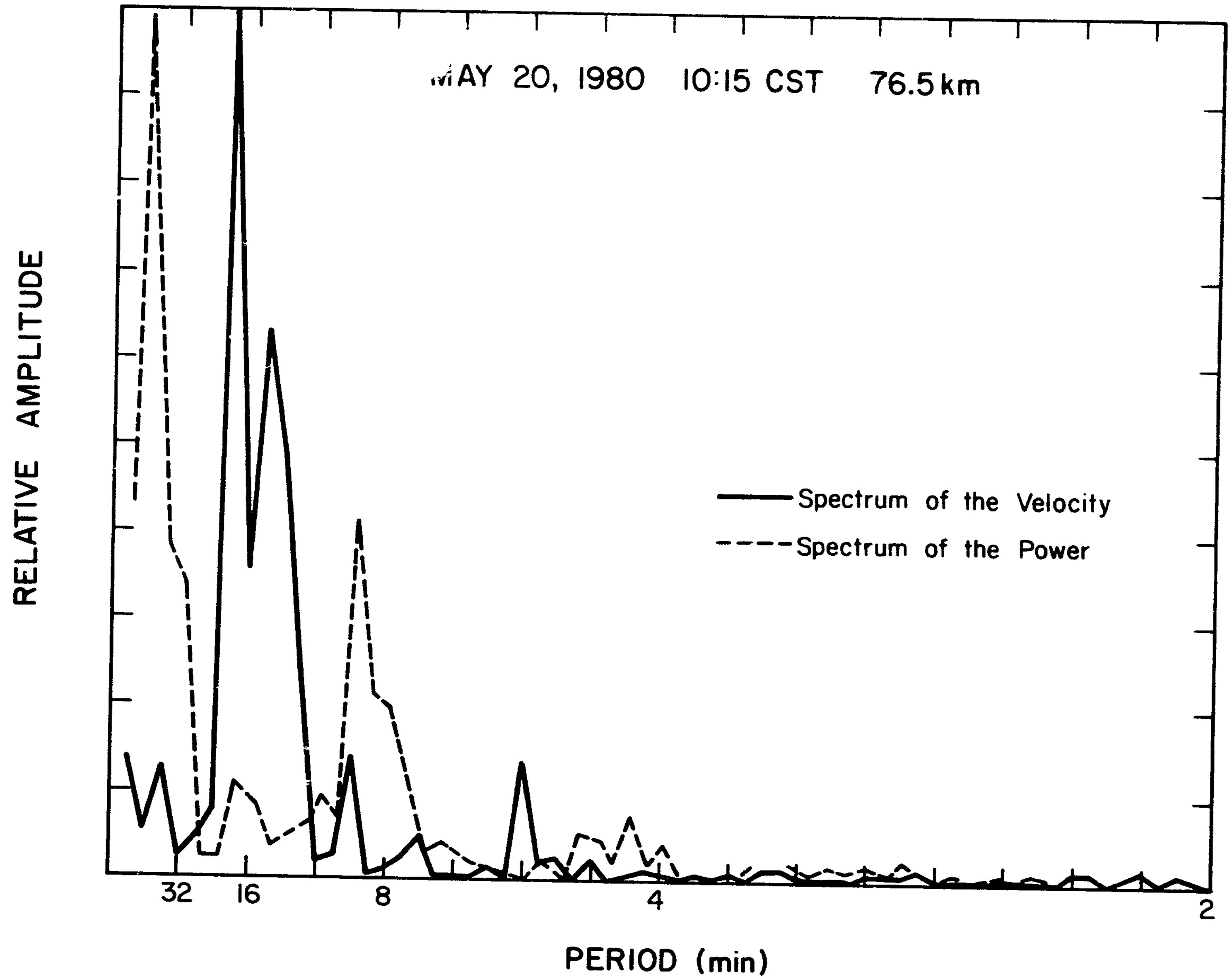


Figure 6.12 Power spectra for the power and velocity variations on May 20, 1980.

spectra at very long periods of one or two hours. Power variations at these time scales, however, are due in part to changing solar flux so no conclusion can be reached. Variations in power are further discussed in the following chapter.

6.5 Scattering in the Presence of Shear

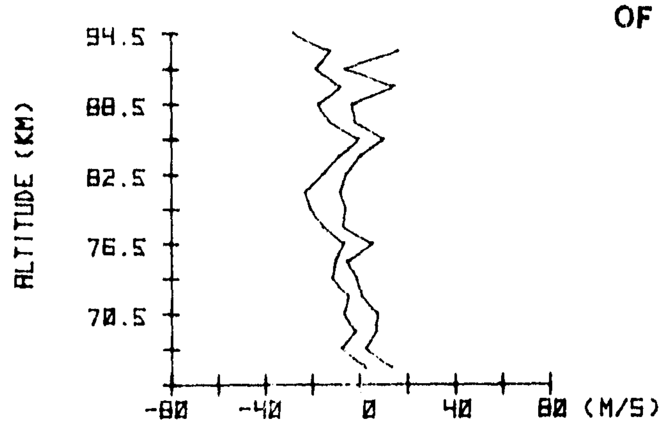
The horizontal velocity toward the northwest which is observed by the Urbana coherent-scatter radar represents a one-hour average of the line-of-sight velocity. The number of minutes for which velocity data are obtained determines the accuracy of the horizontal velocity estimate. On some days the horizontal velocity toward the northwest can be measured over a range of altitudes with sufficient accuracy to indicate a vertical structure and specifically to show regions of shear. Similarly, the vertical structure of the power is illustrated by the hourly statistics. In this section the two sets of hourly statistics are compared in an attempt to observe a relationship between the occurrence of shear and scattering layers.

The profile of horizontal wind which is shown in Figure 6.13 is unusual because of the lack of any significant velocity toward the northwest. Strong scattered power is observed, however, in a layer at 84 km and a second, smaller layer is observed at 76.5 km, however. These layers may be generated by horizontal winds in a direction orthogonal to the horizontal component which can be measured.

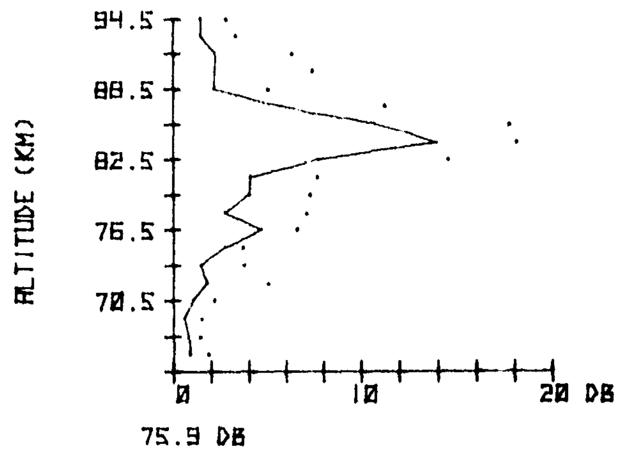
The data shown in Figure 6.14 are more common at Urbana. The horizontal velocity exhibits a quasi-sinusoidal vertical profile which is attributed to tidal motion. The scattering layers appear at regions of low shear which bracket a region of high shear. If a sinusoid and a mean are fitted to the horizontal velocity profile between 69 and 81 km then the scattered power is located near the peaks of the wave.

MAY 23, 1978
60 MINUTE STATISTICS CENTERED ON:

1244 CST



HORIZONTAL VELOCITY TOWARD THE NORTHWEST



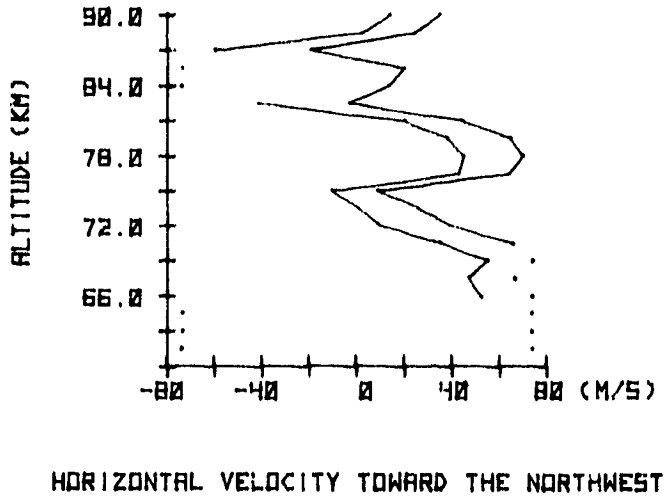
MEDIAN POWER

—— 50%   · · · 90%

Figure 6.13 Hourly statistics for May 23, 1978.

JULY 28, 1982
60 MINUTE STATISTICS CENTERED ON:

1153 CST



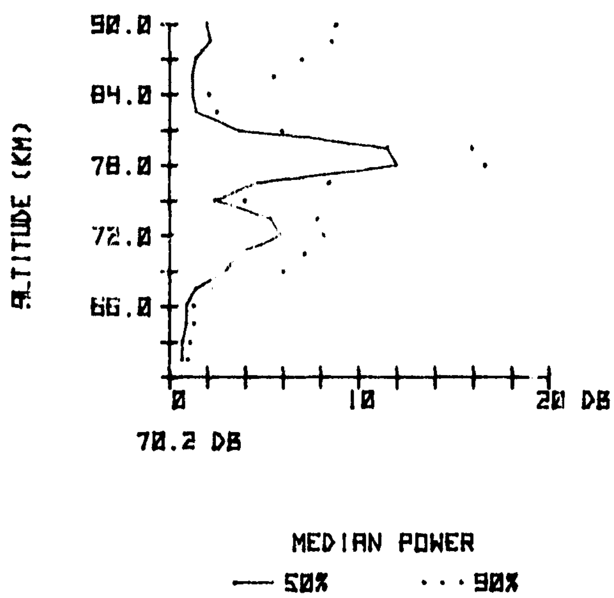


Figure 6.14 Hourly statistics for July 28, 1982.

An unusually strong scattering layer is illustrated in Figure 6.15. The exceptional nature of the velocity data is beyond the plotting capability of the program. However, the data printout indicates that the velocity curves extend linearly downward to approximately 150 m/sec ±10 m/sec at 75 km. It is difficult to assign a wave-like structure to this velocity profile yet it is clear that the region of low shear is near 87 km while the peak in scattered power occurs at 81 km within the region of high shear. The shear in the horizontal velocity does not appear to be the determining factor because the shear is present down to 75 km while the scattered power falls off rapidly below the peak at 81 km.

The variation in the data shown here and within the remainder of the data indicates that no single mechanism determines the altitude at which scattering occurs. If the tidal motion acted alone the regions of high shear would coincide with the power as in Figure 6.15. Yet the power is often observed near regions of low shear as in Figure 6.14. The apparent change in altitude of the scattering with time as observed in daytime data and the nighttime data given above indicates that tides are at least involved in some way. An interaction between the tides and gravity waves either through the generation of locally high shear or via critical-layer activity is indicated.

C-2

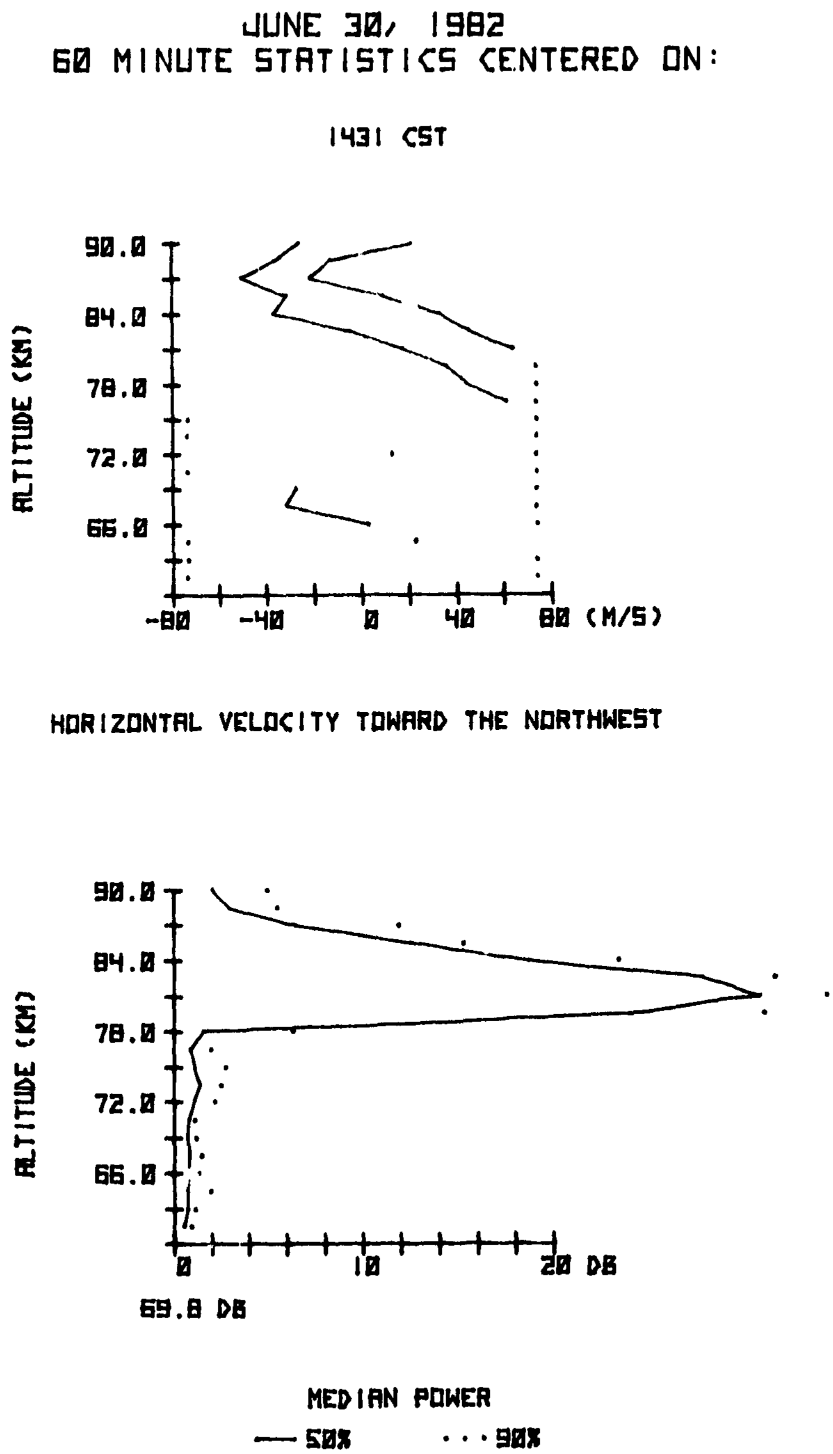


Figure 6.15 Hourly statistics for June 30, 1982.

7. RESULTS FROM THE COHERENTLY INTEGRATED DATA

7.1 Fluctuations in Scattered Power

The variation of scattered power with time has been discussed above in Chapter 2. Recall that variations with time scales on the order of a second represent the changes in the relative positions of turbulent cells and the spinning of turbulent eddies. Variations at longer time scales represent changes at scale sizes comparable or larger than the outer scale of the turbulence. Examples of the two types of variation are given below.

The power data in Figure 7.1 illustrate the variations of power at 10-second intervals for two altitudes on May 19, 1982. Power at the upper altitude decreases continuously with the exception of the burst at 12:41:10 CST. The altitude and duration of this burst indicate that it is due to a meteor. The slow decrease in power is due to a decrease in the amount of turbulence in the beam at 82.5 km.

While the scattered power at 82.5 km is consistently above the estimated noise power, at 69 km the power is significant for only 2 minutes. This enhancement is clearly not due to one or even several meteors because of its duration and low altitude. Furthermore, an enhancement in ionization, which generally occurs at all altitudes simultaneously, is not indicated in Figure 7.1. The 2-minute change in power, therefore, corresponds to the passage of a turbulent region through the beam of the radar. At an altitude of 69 km the beam is 3.6 km wide so that an object of negligible horizontal dimension is advected through the beam in 2 minutes at 30 m/sec. If the layer has a horizontal extent of 1 km the required speed is 38 m/sec. These values of horizontal velocity are often measured at Urbana.

The presence of one-minute bursts in the minute-by-minute data is, therefore, due to two phenomena. At low altitudes the one-minute burst

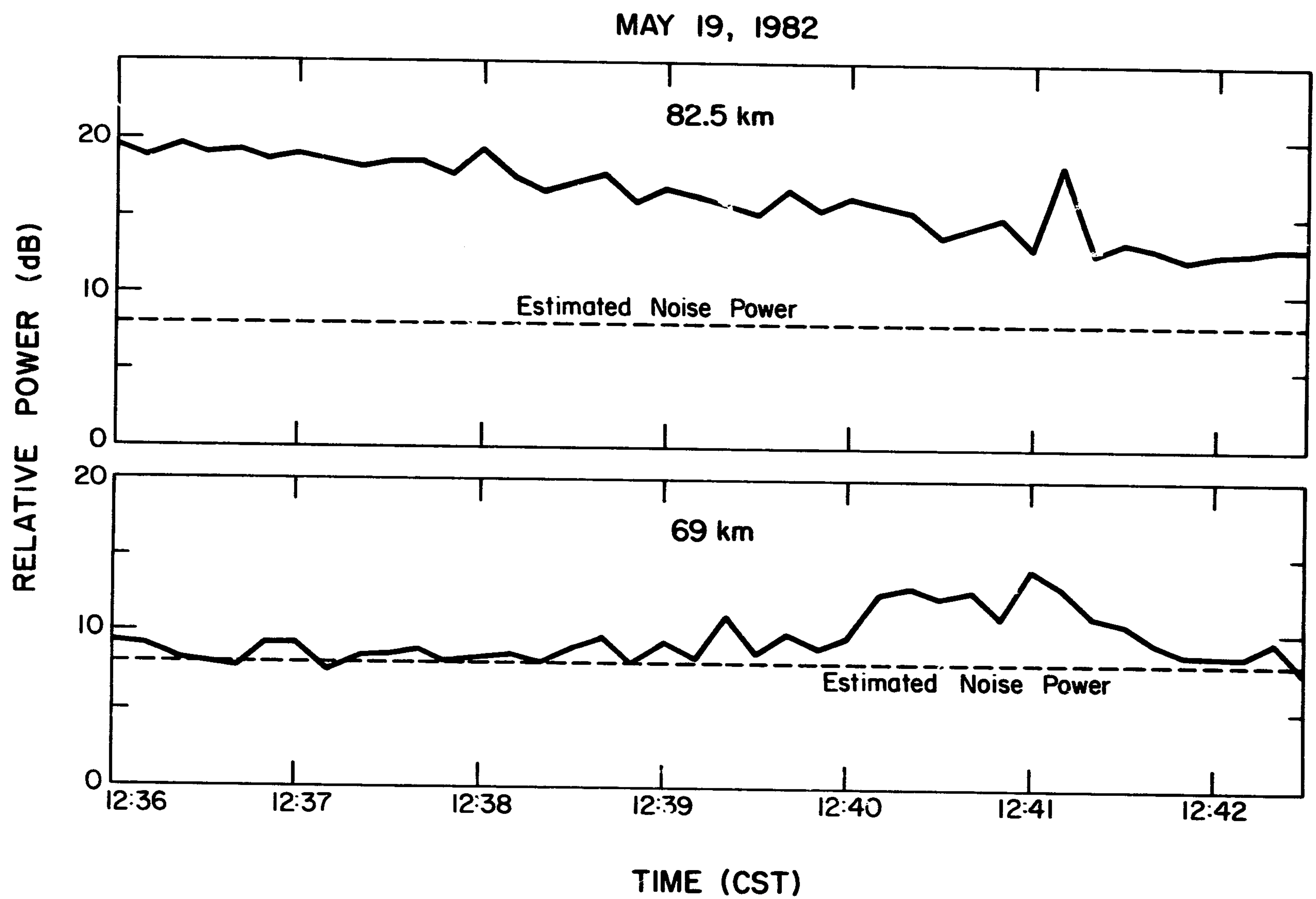


Figure 7.1 Power data at 10-second intervals for 69 km and 82.5 km on May 19, 1982.

corresponds to an actual increase for a period of one to two minutes and represents the advection of turbulence. This is particularly true if the one-minute burst rises out of the noise power. At higher altitudes a short but powerful burst due to a meteor also appears as a one-minute burst in the routine data. The observation of power at 10-second intervals as shown here appears to be necessary to unambiguously identify meteors.

A section of the data shown in Figure 7.1 is expanded in Figure 7.2 with the power summed for only one second. This time scale is near the correlation time of the scattered signals. The characteristic shape of a meteor return is again observed at 12:41:10 CST--the sharp increase followed by a slower decrease. Note that the fluctuations at 69 km in Figure 7.2 are confined to lower frequencies relative to the fluctuations at 82.5 km. This variation of the power spectra with respect to altitude is discussed below.

7.2 Altitude Variation of Power Spectra

Several of the coherently integrated data sets show scattering at more than one altitude. The data shown in Figure 7.3 represent a typical variation of the spectrum with altitude. The spectrum at 69 km indicates a single scattering region within the beam while the spectrum at 82.5 km resembles the spectrum generated from filtered random data which was shown in Chapter 4. This leads to the conclusion that at the higher altitudes the beam contains many scatterers.

The profile of relative power observed at Urbana is given by Royrvik et al. [1982]. The 90% relative power increases from 63 km to 81 km at a rate of about 0.28 dB/km. For example, the power at 82.5 km is about 4 dB higher than the power at 69 km in Figure 7.3. The opposite situation is illustrated in the spectra shown in Figure 7.4. The power at 72 km is 10 dB above the noise and the spectrum is somewhat wider than the 69 km example of

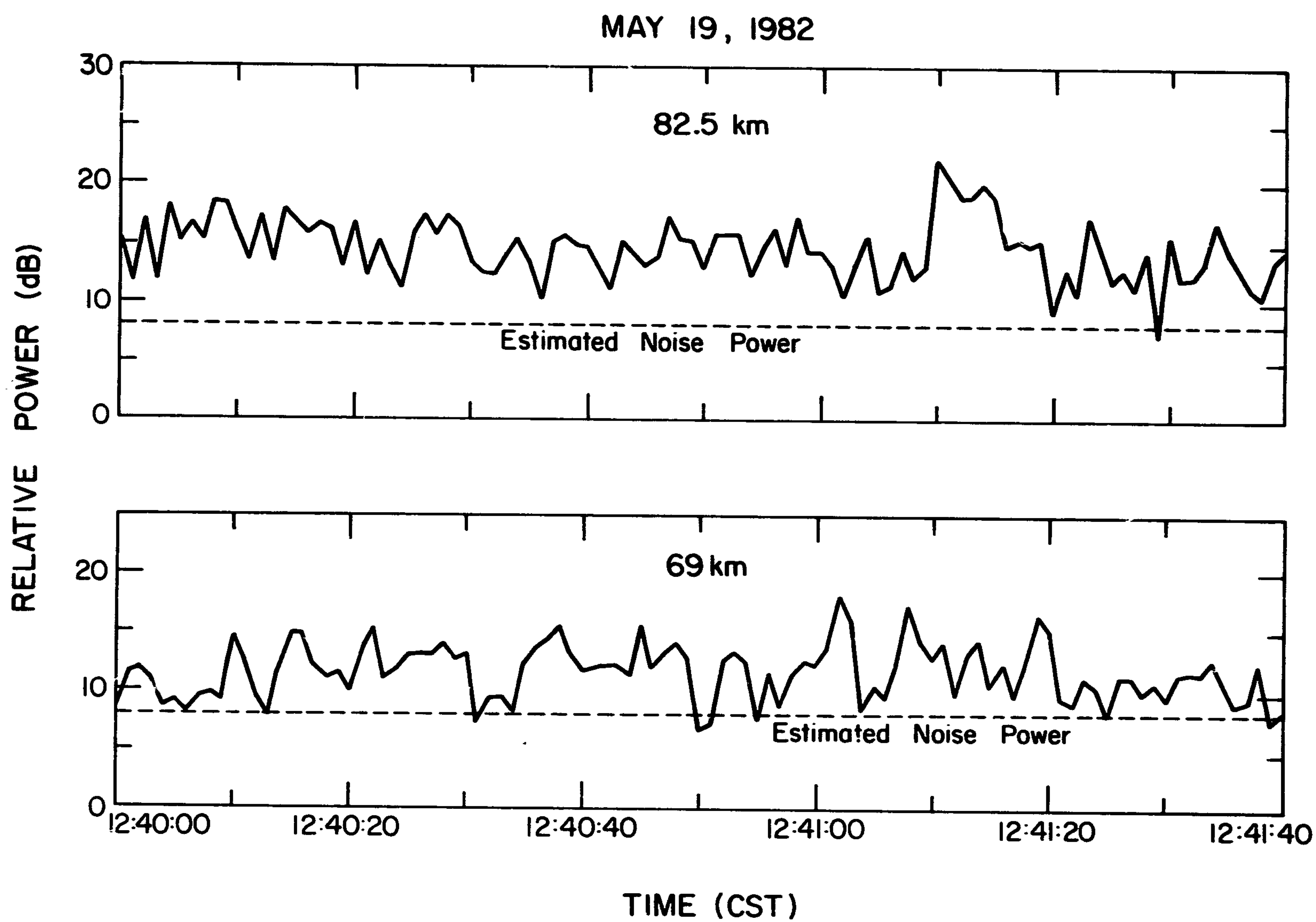
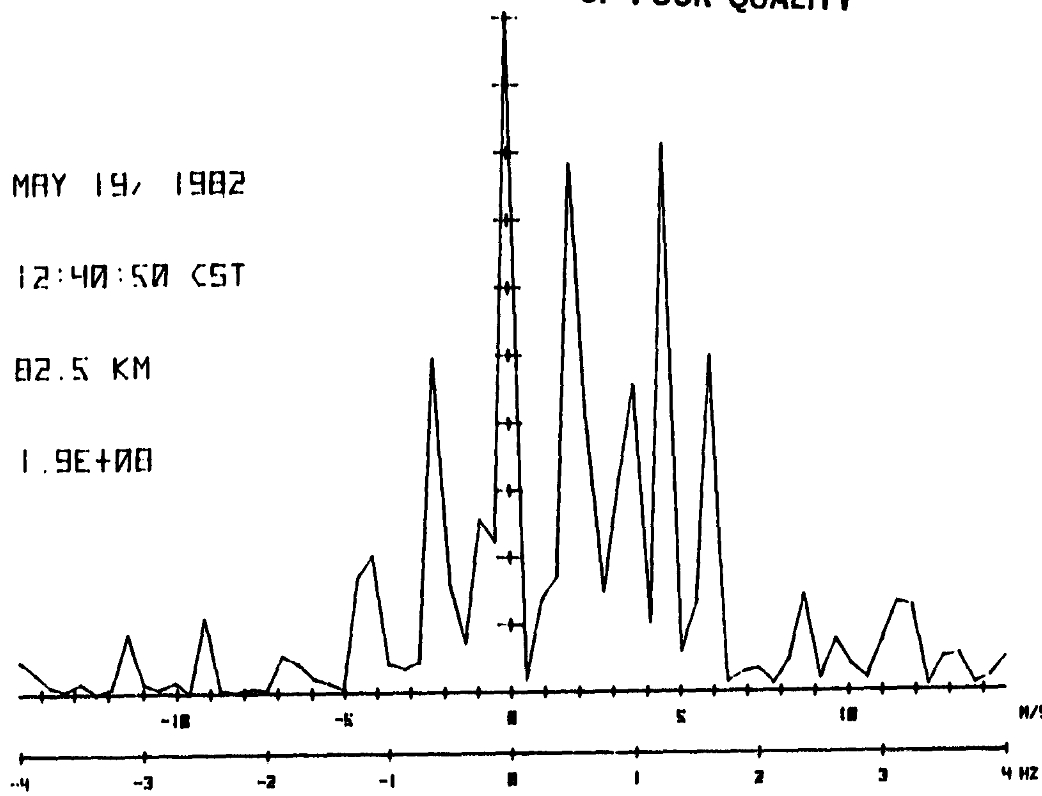


Figure 7.2 Power data at 1-second intervals on May 19, 1982.

ORIGINAL PAGE IS
OF POOR QUALITY



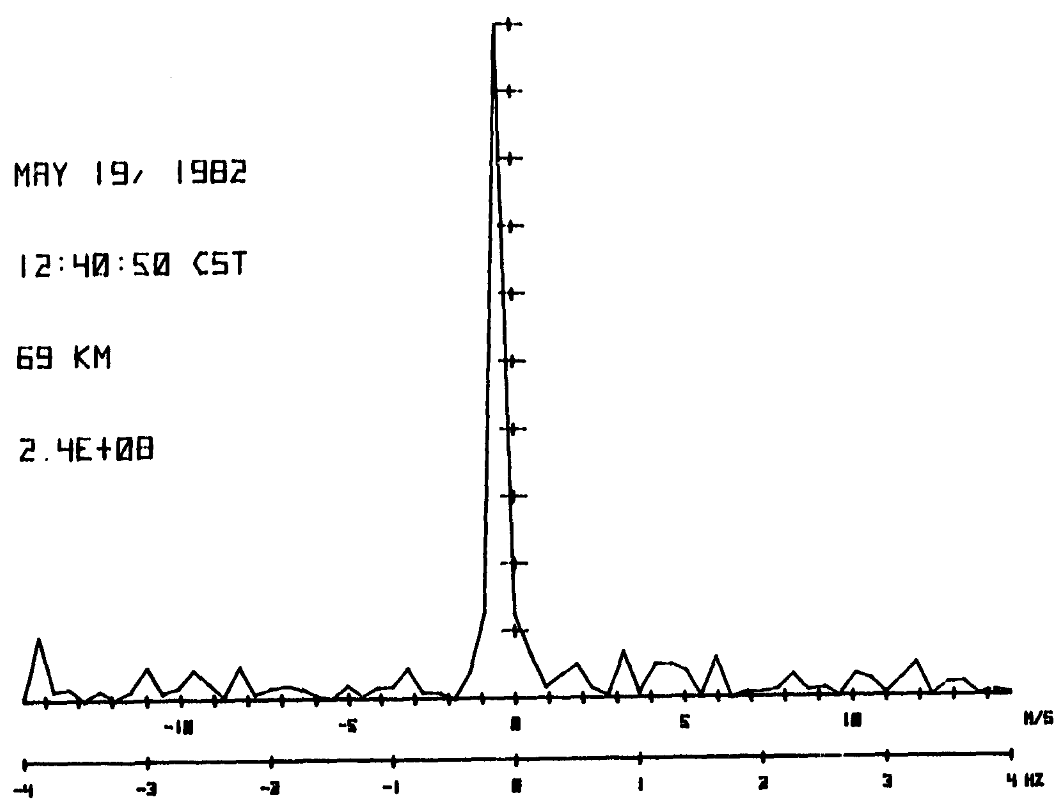


Figure 7.3 Spectral variation with altitude on May 19, 1982.

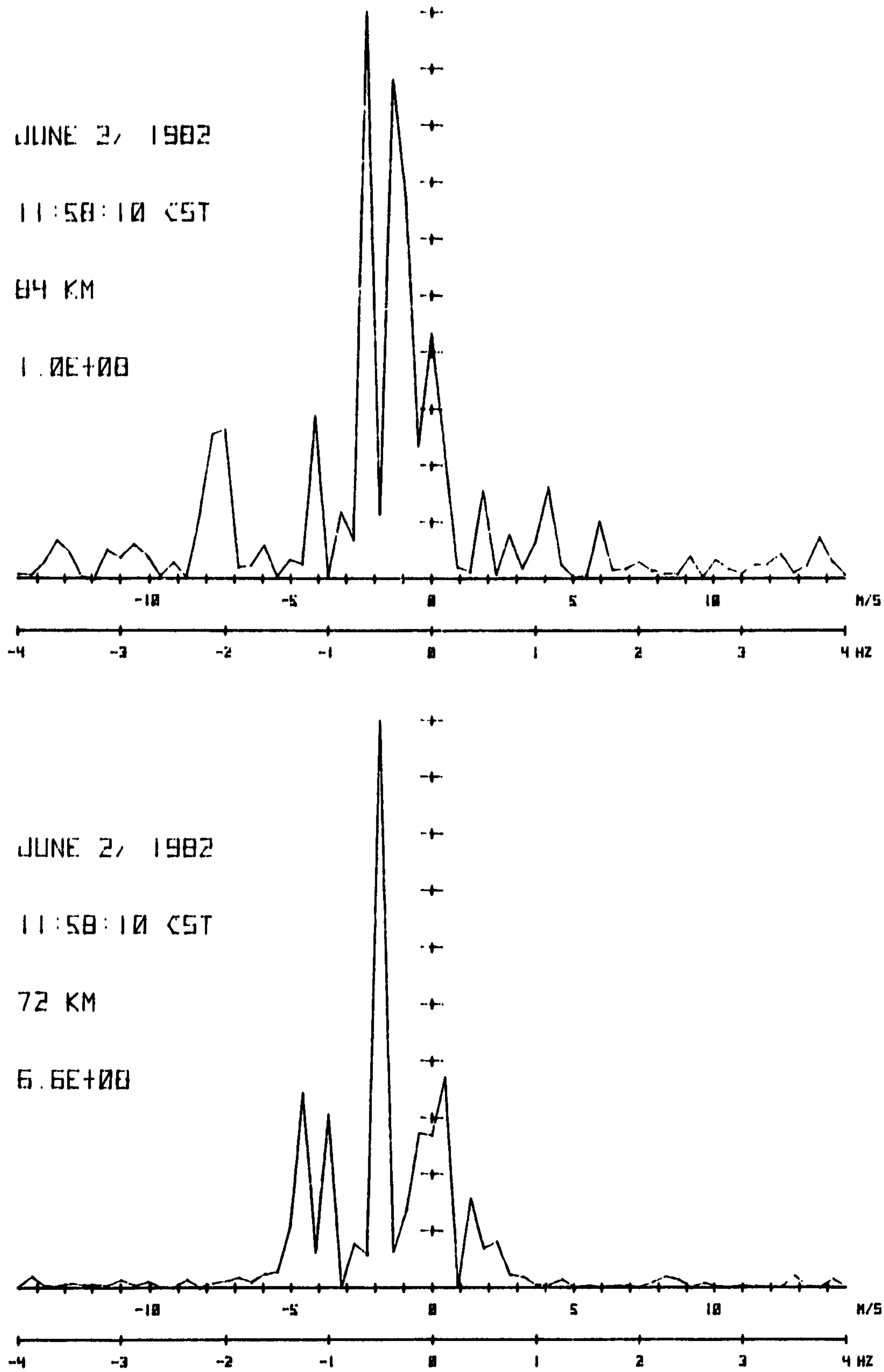


Figure 7.4 Spectral variation with altitude on June 2, 1982.

Figure 7.3. If the noise powers at the two times are equated then a 7 dB increase in power occurs from Figure 7.3 to Figure 7.4 at low altitudes while the power at the higher altitudes is roughly the same. An increase in spectrum width with increasing power is indicated at low altitudes for these two examples.

A counterexample is illustrated in Figure 7.5. Here the power level at 70.5 km is 13 dB above the noise--an unusual amount of power for this altitude. Yet the spectrum is narrow with the characteristic shape of one or two distinct frequencies. The power at 85.5 km is also high at 17 dB above the noise. It is possible that an enhancement of the ionization led to the increase at both altitudes. To effectively study the relationship between spectral width and scattered power the effect of changes in ionization level must be removed.

In the three examples presented here the spectra above 72 km are wider than those at or below 72 km. This is true of all of the coherently-integrated data sets. Even for those days when only a single scattering region is observed the spectrum is narrow if at a low altitude and wide if at a high altitude. Similar observations are noted by several authors; e.g., Royrvik et al. [1982], and Rottger et al. [1979]. Fukao et al. [1980] and Countryman and Bowhill [1979] also show anisotropy in the scattering below 72 km. The anisotropy and the observation of longer correlation time associated with increased power by these authors has led Rottger and Liu [1978] and Gage and Balsley [1980] to consider diffuse or Fresnel scattering to contribute at the lower altitudes. The model is one of turbulent eddies of half the radar wavelength randomly distributed throughout the beam at higher altitudes but at lower altitudes arranged in vertically thin, horizontally broad layers. Gage and Balsley [1980] point out that anisotropic

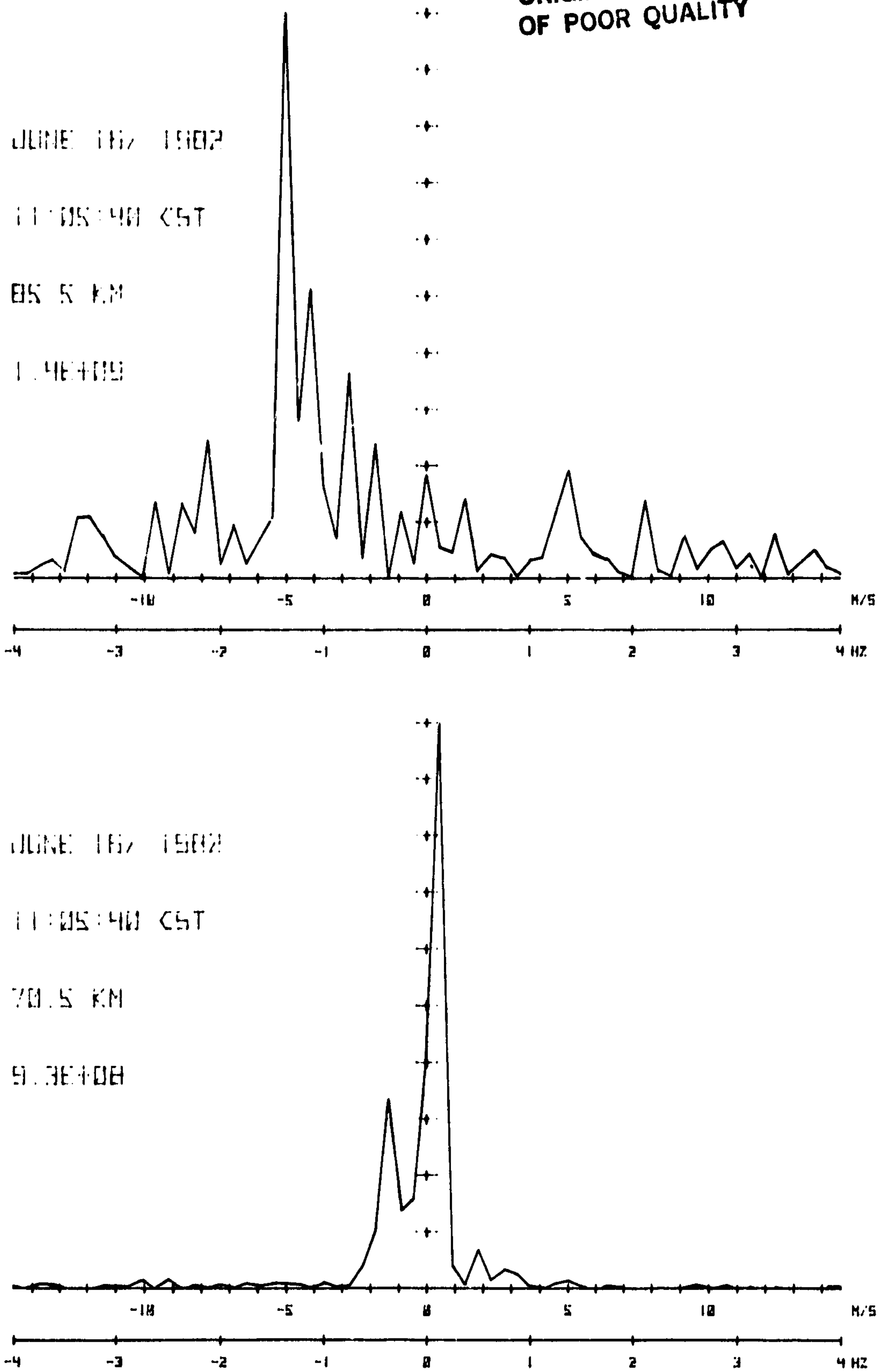


Figure 7.5 Spectral variation with altitude on June 16, 1982.

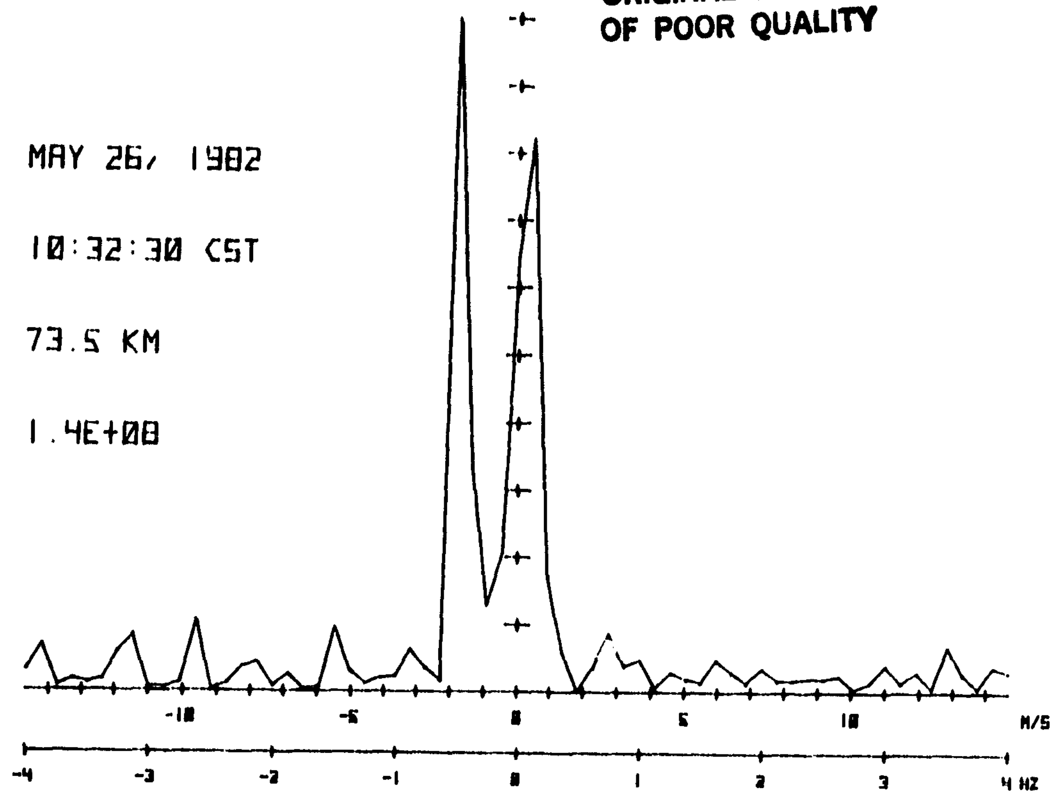
turbulence may be indistinguishable from Fresnel scattering, however. Certainly the data shown here support some aspects of the model given above.

7.3 Use of Doppler Information to Increase Range Resolution

The spectra given above for low altitudes in the mesosphere typically show one or two peaks at distinct Doppler velocities. As discussed in Chapter 4, these peaks are interpreted as distinct scattering layers within the scattering volume. The specific altitude of each scattering layer is calculated and plotted, providing a higher resolution picture of the scattering volume than can normally be obtained at Urbana. Several examples are presented below.

If the received signal is oversampled in altitude then a return from a specific scatterer will appear at adjacent altitudes. The relative contribution at each altitude will depend on the pulse shape of the transmitted signal, the impulse response of the receiving system, and the position of the scattered signal with respect to the sample gates. The width of the scatterer is assumed to be narrow with respect to the pulse width and so have minimal effect on the time response. Variations in the phase of the signal from pulse to pulse will not affect the relative amplitude factor during the time that the altitude of the scatterer remains constant. Power spectra of short duration at adjacent altitudes will therefore show significant power at the same Doppler frequency. If the transmit and receive characteristics are known then a comparison of the power at the Doppler frequency in the adjacent spectra permits the specific altitude to be determined.

Figure 7.6 illustrates spectra at adjacent altitudes with a contribution from the same scatterer. The +1/8 Hz component at 73.5 km is 62% of its value at 72 km and therefore is located closer to 72 km. To determine the



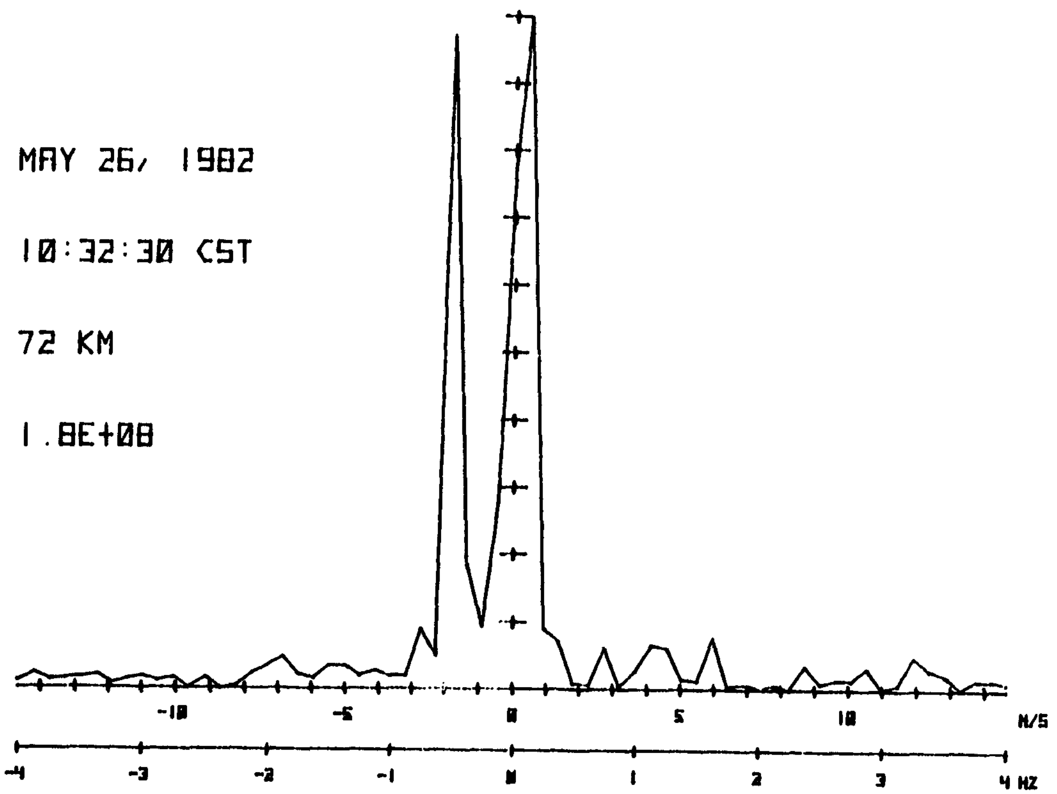


Figure 7.6 Comparison of spectra at adjacent altitudes.

specific altitude a parabola is fitted to the amplitude of the Doppler frequency at the altitude of its maximum and the first altitude above and below the maximum [Backof and Bowhill 1974]. The peak of the parabola is taken as the altitude of the scatterer. Specifically, if the amplitudes at the three adjacent altitudes in order of increasing altitude at A_1, A_2, and A_3 with 10 μs spacing between samples then the offset τ in μs from altitude 2 is given by

$$\tau = 5\left[\frac{A_1-A_3}{A_1+A_3-2A_2}\right] \tag{7.1}$$

The offset is converted to range and added algebraically to the real altitude corresponding to altitude 2.

The method given above produces a reasonable approximation to τ even if the characteristic pulse shape is distorted by finite thickness of the scatterer. The modeling of the pulse as a parabola produces a maximum error of 1 μs for a Gaussian pulse and therefore a range accuracy of ±150 m. This performance has been verified by Harrington and Geller [1975] for low signal-to-noise ratios. The presence of an additional scatterer at the same Doppler frequency within the actual 3 km range resolution of the radar also produces an error in the altitude estimate. The error will be in the direction of the second scatterer because the amplitude at the altitude on that side of the maximum sample will be increased. In general the scattering layers are sufficiently separated in altitude and Doppler frequency to avoid this error.

The horizontal winds and the gravity waves produce a velocity which changes slowly with respect to the time interval between spectra. The peaks in the spectra can be followed in time with a contour map as shown in Figure 7.7. Only the upper two contour levels have been shown to emphasize the peaks. The peaks are near 0 Hz at 09:38:00 CST, move to roughly 1.25 Hz and

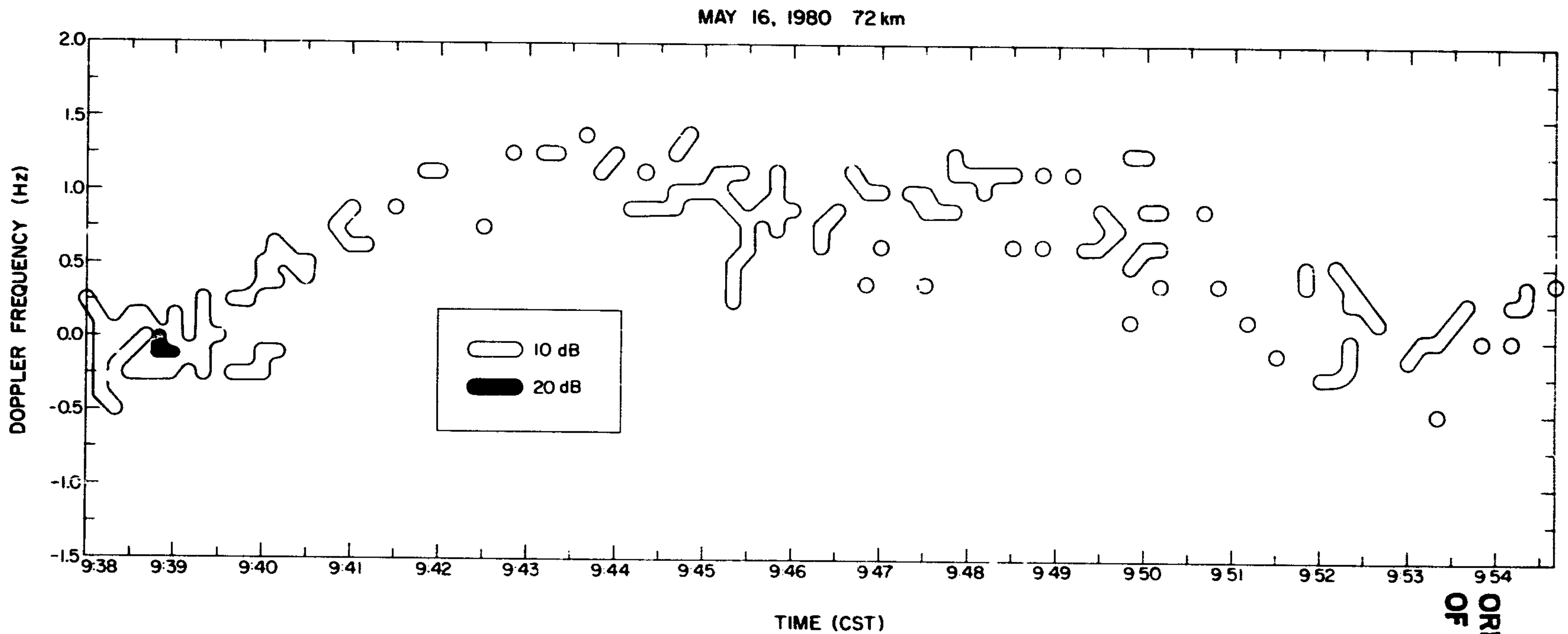


Figure 7.7 Contour map of power vs. Doppler frequency and time.

then back to 0 Hz by 09:54:00 CST. Assuming that the two endpoints of the data set represent wave minima yields a wave of roughly 16 minute period and 3 m/sec amplitude in conjunction with a horizontal velocity of about 70 m/sec. If the endpoints are not the minima the period and amplitude of the wave increase and the horizontal velocity is reduced. A comparison of the contour maps at adjacent altitudes permits identification of the peaks in the spectra which corresponds to scatterers. The altitude of each peak is plotted as shown below.

The data presented in Figure 7.8 represent the first 6 minutes corresponding to the contour of Figure 7.7. The size of a circle is proportional to Doppler frequency while the center of the circle indicates the altitude. Negative frequencies corresponding to upward velocities are indicated by filling in the circle. A scattering region near 71.6 km changes from positive Doppler to negative Doppler in 2 minutes and then disappears. At higher altitudes 3 regions are indicated: a short lived region at 72.4 km, a second region near 73 km from about 09:40:00 to 09:41:00 CST and a third region at 72.4 km. It should be noted that the 71.6 km scattering layer moves slowly upward in general agreement with the small negative Doppler velocity indicated from 09:38:50 to 09:40:10 CST. The upper layers do not appear to change altitude in agreement with their indicated line-of-sight velocity. This discrepancy indicates a significant contribution to the line-of-sight velocity by the horizontal velocity.

The horizontal velocity is also significant in Figure 7.9. A negative Doppler velocity, corresponding to an upward motion, is observed throughout the first 5 minutes of this data set yet only at the lowest altitudes is a consistent upward motion indicated. The average change in altitude of the layer which begins at 68.7 km is roughly 1.8 m/sec. The remainder of the

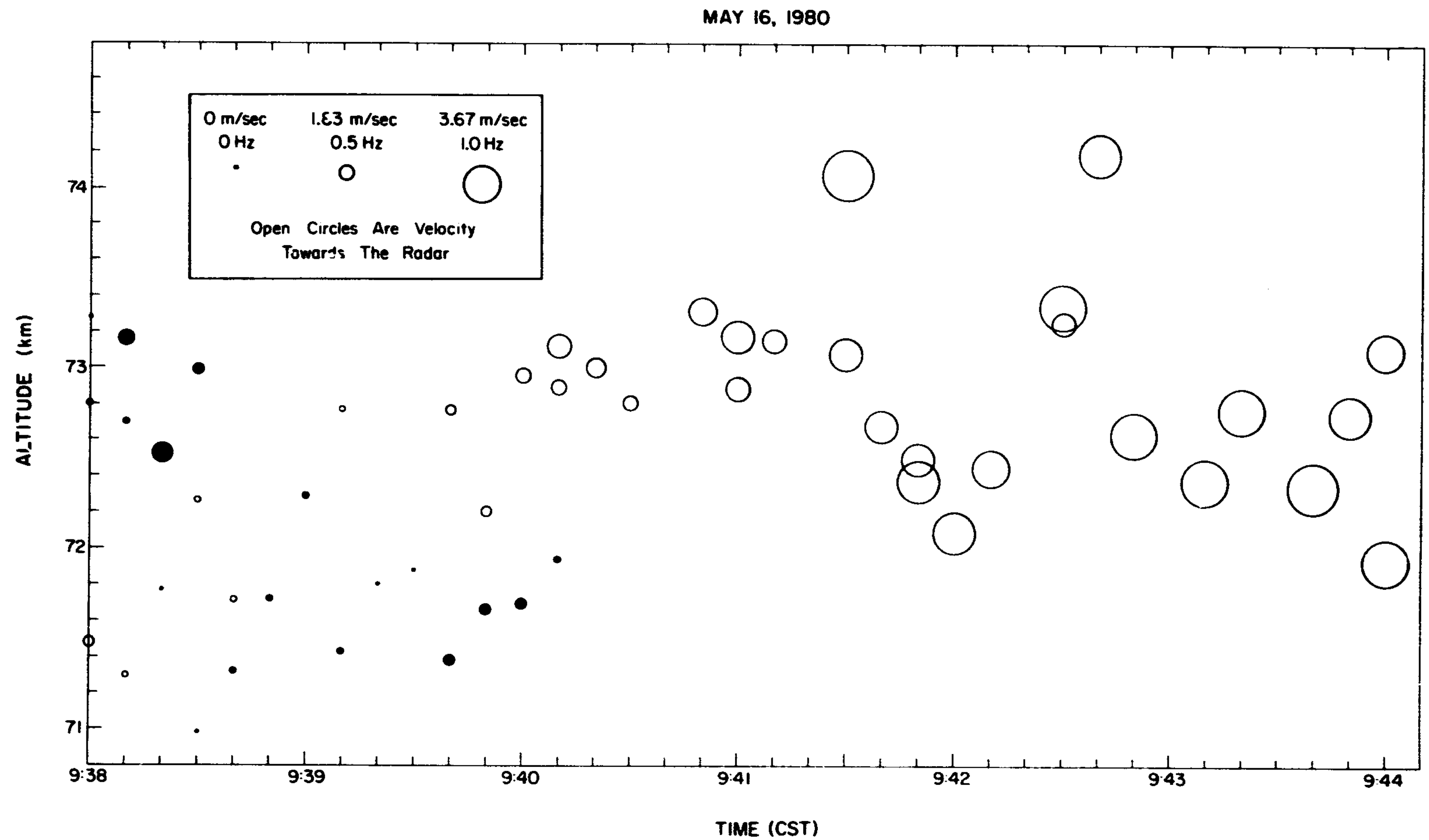


Figure 7.8 Scattering on May 16, 1980.

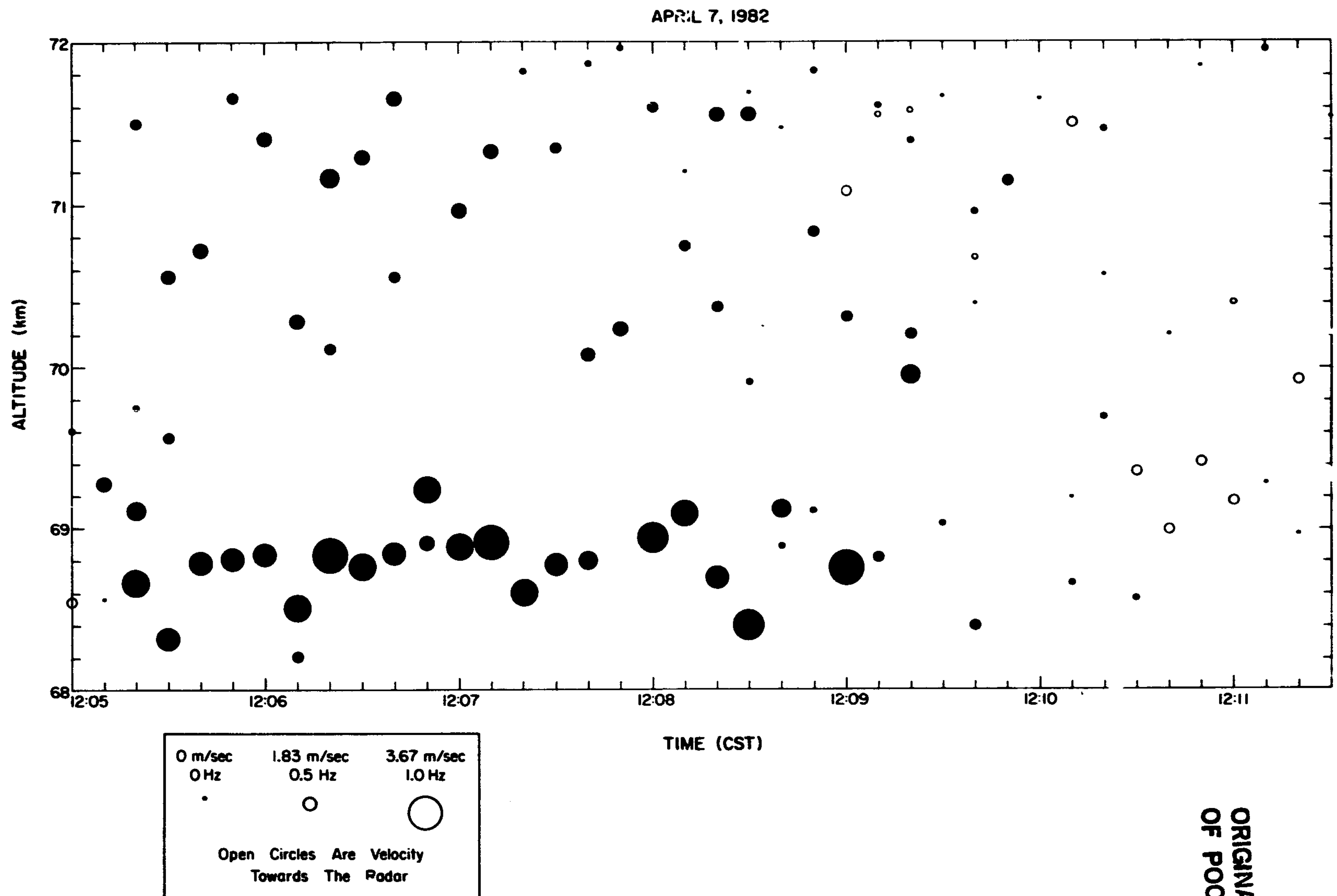


Figure 7.9 Scattering on April 7, 1982.

negative Doppler frequency is associated with a horizontal velocity of approximately 50 m/sec. The scattering from the higher altitudes in Figure 7.9 is not as easily analyzed, however. There appears to be two separate layers, one at 70.4 km and the other at 71.5 km. The 70.4 km layer is present for only a short time between 12:05:30 and 12:06:40 CST but reappears at 12:07:40 CST and continues intermittently throughout the remainder of the data set. There is an apparent downward motion of this layer which is consistent with the observations in the following way. The horizontal velocity at 70 km should be similar to that at 68 km, i.e., it should produce a negative Doppler frequency. The small positive Doppler associated with the vertical motion adds to this with the net result being a Doppler frequency less negative than that at 68 km.

In both Figures 7.8 and 7.9 scattering layers appear to change altitude together at the start of the data set. This variation must be examined in light of the error in determining the altitude. The accuracy of the altitude estimate using the parabolic fit is a worst-case error of ± 150 m in the absence of noise. This accuracy degrades gracefully with noise and pulse-shape distortion as discussed above [Backof and Bowhill 1974]. Therefore, the error is large enough to remove most of the observed variation. However, the error should be random while the variation from a mean altitude does not appear to be random. The variation in Figure 7.8 is slow enough to be attributable to a gravity wave but the variation in Figure 7.9 occurs at an acoustic-wave period. In both cases the presence of two closely spaced layers may be an indication of a Kelvin-Helmholtz instability. Recall that turbulent eddies are generated on both sides of the initial location of the shear. Whether the turbulent layers appear in pairs on a regular basis to support this idea requires further investigation.

Finally, consider the example of a short burst of power which is illustrated in Figure 7.10. The scattering layer moves through the beam in one minute but during that time the Doppler velocity changes from about +3.5 m/sec to near -2.5 m/sec. This variation would be averaged to a near zero value in the minute-by-minute data. The 10-second interval between spectra is, therefore, necessary to study this rapid variation which may also be the result of acoustic waves.

ORIGINAL PAGE IS
OF POOR QUALITY

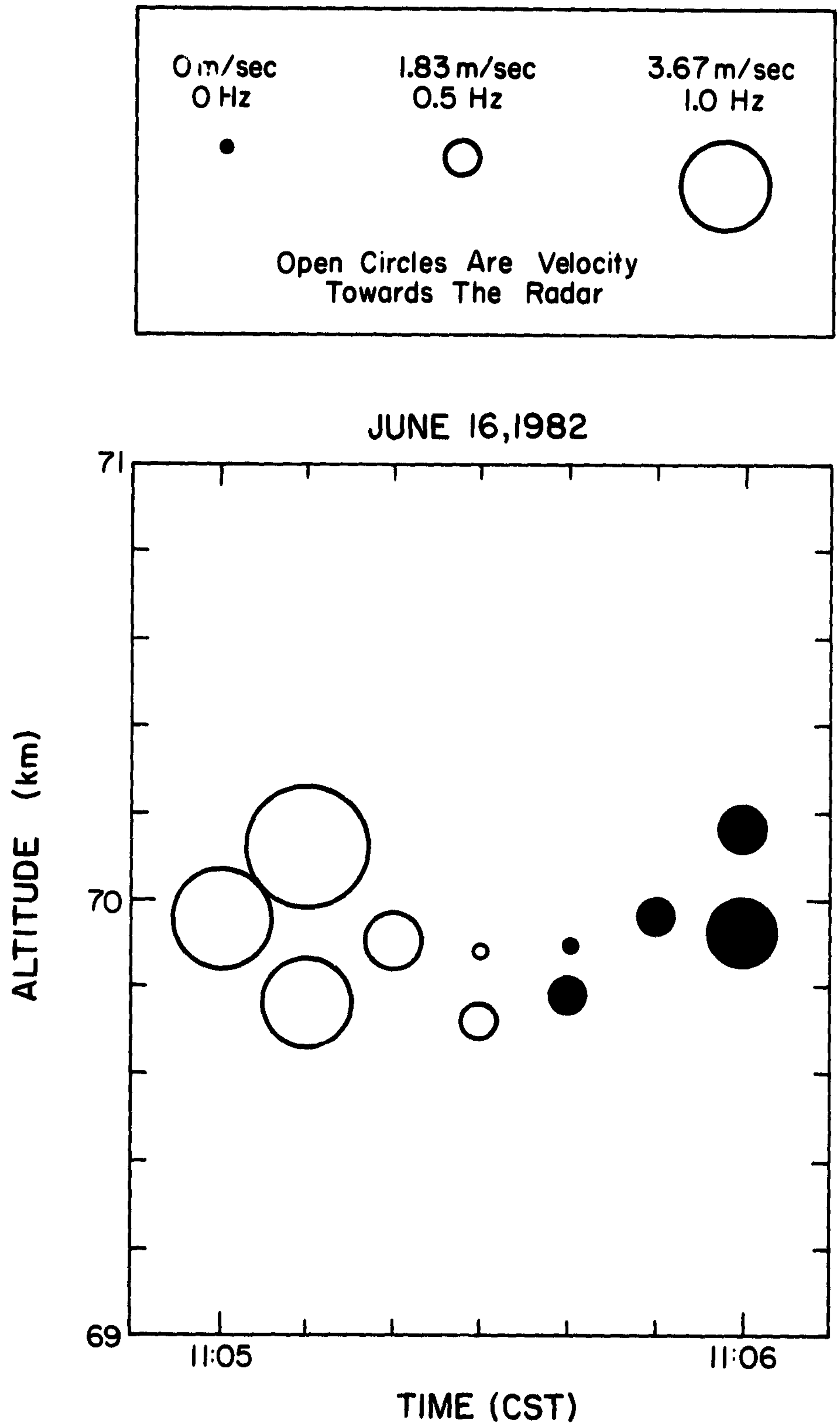


Figure 7.10 Scattering on June 16, 1982.

8. SUMMARY AND SUGGESTIONS FOR FUTURE RESEARCH

8.1 Summary

The principal conclusions of this study are summarized below.

1. The variation in ionization produced by changing solar flux causes a similar change in the signal-to-noise ratio of the coherent scatter.

2. Under conditions of enhanced ionization, either due to unusual solar activity or to meteor showers, coherent scatter can be observed at night with sufficient continuity to observe wave motion. The nighttime mesosphere shows gravity wave motion and an apparent vertical motion of a scattering layer similar to the daytime data.

3. Limited time periods show some correlation between the variation of scattered power and the velocity. However, the spectra of the two types of data are typically unrelated at gravity-wave frequencies. This implies that, in general, the gravity waves do not generate and transport turbulence independent of the horizontal wind.

4. Examples of turbulent regions at altitudes with large shear in the horizontal wind and at altitudes with small shear indicated that the horizontal winds also do not act alone to produce the turbulence.

5. A ten-second time resolution is necessary for the identification of meteors, to study the short bursts of power which occur at low altitudes and the rapidly changing Doppler velocities which may indicate acoustic waves.

6. The data at Urbana show a difference in spectral width above and below 75 km. The narrow spectra below and wider spectra above this altitude agree with the model of bulk scattering at higher altitudes and Fresnel scattering at the lower altitudes.

7. Doppler-frequency information can be used to improve the range resolution of the coherent-scatter experiment.

8. The occurrence of pairs of scattering layers may be evidence for the generation of turbulence in the mesosphere by Kelvin-Helmholtz instability.

8.2 Suggestions For Future Research

Several changes in the collection and processing of coherent-scatter data are suggested by the work above. It is clear that additional data with a time resolution on the order of ten seconds should be collected to study rapid changes in the spectra. Fukao et al. [1980] show that the spectrum width decreases when a shorter time series is used to calculate the spectrum. The spectra presented in this work show that at one-minute time resolution the width of the calculated spectrum is increased by the slow change in the center frequency of the instantaneous spectrum due to bulk motion. If the time interval is too short, however, the variance in the estimate of the spectrum increases. Furthermore, the frequency resolution of the spectrum will decrease if the number of sample points is reduced. The time interval must, therefore, be chosen using a matched-filter concept based on the temporal variation of the scattering.

The data with ten-second resolution which are shown in this work do not illustrate any of the sudden increases in power observed at high altitudes in the one-minute data. This points out a need for the collection of the higher resolution data for long periods of time. These bursts of power, believed to be due to the breaking of waves, are unlike those at lower altitude primarily because of their longer duration. High resolution spectral information may answer questions concerning the source of the sudden increase in scattering and the gradual decay over periods of up to 15

minutes.

Continuous collection of coherently-integrated data for long periods will also allow further investigation of the reflection versus scattering question. Rottger [1980] discusses many techniques for the study of this question: wavelength dependence, volume dependence, aspect sensitivity, temporal variations, Doppler spectra, phase distributions, amplitude distributions, temporal correlation of the signal intensity, spatial correlation and dependence of signal power and correlation time. Because of the limited height resolution at Urbana the most promising technqiues are the amplitude distributions and the temporal correlation of the signal intensity. Rottger [1980] gives an example from the troposphere of an amplitude distribution which indicates a mixture of scattering and reflection. The amplitude distribution approaches a Rayleigh distribution for scattering and the Dirac delta for reflection. The intensity correlation function can also be used to determine relative amounts of scattering and reflection because it depends only on the relative motion of the scatterers. The data shown in this work indicate that several layers often appear within the scattering volume at lower altitudes however, so that measurements of the intensity correlation will include the interaction between multiple layers.

If the range resolution of the radar is improved through pulse coding the multiple layer problem can be reduced or eliminated. The volume dependence technique discussed by Rottger [1980] would then be used to investigate reflection or Fresnel scattering from very thin layers below 75 km. The power from a thin layer will remain constant as the vertical size of the scattering volume is increased. The returns from bulk scattering however, will also show increasing power with increasing vertical size. The range

resolution could be further improved by the use of Doppler information as in this work. The investigation of pairs of layers which appear to move together would also be enhanced by improved range resolution.

The collection of large amounts of data with ten-second resolution will require changes in the experimental procedure. One relatively inexpensive way to collect and store coherently integrated data is via a microcomputer and magnetic-tape system. The process of coherent integration could be performed in real-time and the data stored on magnetic tape in a standard format. Postprocessing of a magnetic tape would be done on a large mainframe, a minicomputer, or a second microcomputer with a compatible tape drive. As in this work, the specific times and heights of interest would first be identified by calculating the power in the scattered signal. The calculation of the spectra would occur at only the selected times and altitudes.

The generation of turbulence appears to be the result of an interaction between the horizontal winds produced by tides and the vertically propagating gravity waves. This investigation has been limited by the absence of a complete horizontal-velocity vector with sufficient altitude resolution. As noted above, this limitation leads to the observation of turbulence in the absence of indicated shear. If the turbulence is due to shear then the shear must be in the direction orthogonal to the measured horizontal velocity or at a smaller vertical scale than can be measured with the present range resolution. The small vertical scale would be expected to correspond to the generation of closely spaced layers. Improvement of the antenna and related equipment would permit beam-swinging to obtain two horizontal velocity measurements and would also allow the study of the aspect sensitivity of the returns.

The lowest level of scattered power in a given minute is assumed to be the noise level in the present equipment at Urbana. A more accurate comparison of power from day-to-day could be made if samples of the noise were collected. The transmitter would be turned off and the noise samples coherently integrated in the same manner as the signal samples. The resultant noise power could then be subtracted from the data. In addition, a log of the noise power would serve as an indication of the performance of the receiving system. A similar day-to-day performance comparison should be made using a reference oscillator which generates a known Doppler frequency and amplitude. The reference oscillator should be synthesized from the radar frequency source in order to reduce problems with thermal drift. A computer program would be used to verify radar performance before beginning data collection.

The calculation of the power spectra for the minute-by-minute velocity and power data should be modified to permit calculation for time periods of less than two hours. This will permit further comparison of the variations in scattered power and velocity. Based on the data examined for this work a 30-minute time period would reveal an increased correlation in the variations. The use of zeroes for each minute when no velocity is determined can produce excessive noise at high frequencies in the spectrum. In general this noise prevents observation of the Brunt-Vaisala barrier. By proper filtering of the data a study of the variation in the Brunt-Vaisala frequency could be carried out with the present data base of minute-by-minute data.

Finally, the variation in the width of the spectra with changing altitude indicates that a variable coherent-integration time is appropriate. This could be implemented by adding a different number of samples in

different altitude regions. For example, the coherent-integration time below 75 km might be 1/4 sec while above 75 km it would reamin 1/8 sec. The resolution of a spectrum calculated over a fixed time period (i.e., the same for all altitudes) is not affected because the sampling frequency associated with the coherently integrated samples and the number of samples in the fixed time period are proportional. The format of the data tapes should include information on the coherent-integration time to aid in postprocessing. The calculation of the optimum coherent-integration time could be done in real-time using adaptive-filter techniques. However, it is likely that after a short time fixed coherent-integration times could be chosen with a small loss in performance. The use of coherent-integration times matched to the data will increase the signal-to-noise ratio and potentially allow the observation of more data at low mesospheric altitudes.

APPENDIX

COMPUTER PROGRAMS

A.1 Introduction

The programs discussed in this appendix fall into two broad categories based on the data base with which they operate. The first group of programs in Sections A.2 to A.8 deals with the coherently integrated data. The remainder of the programs work with the minute-by-minute data or with data derived from the minute-by-minute data. The majority of these programs are written in BASIC for the Apple computer which is the repository for the coherent-scatter data. The programs below are those which are specific to this work. However, they represent a logical extension of the programming effort by this author, published by Roth [1982], which transferred the focus of the postprocessing from the HP-9830 to the Apple II. These programs demonstrate the usefulness of a microcomputer in a research environment.

A.2 Collection Programs for the Coherently Integrated Data

The collection of coherently integrated data is very similar to the routine data collection described by Gibbs and Bowhill [1979]. Each disk of the PDP-15 is filled with 400 seconds of data in a single file. The file consists of 40 records of 10 seconds of data for 20 altitudes. The coherent integration occurs over 50 radar pulses using 25 samples of the cosine channel and 25 samples of the sine channel. Because the PRF of the radar is 400 Hz the coherent-integration time is 1/8 second.

The collection programs SCAT and CM are shown in Tables A.1 and A.2, respectively. They are similar to the HSCAT and DM programs normally used except that SCAT does not autocorrelate in real time and the buffer length in CM is changed. From the operators perspective the difference is primarily that coherently integrating without autocorrelating fills 3 disks

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.1 SCAT.

```
        .GLOBL  MAIN,CM,.DA
        .IODEV  5,6,7
MAIN    DZM     SEQ#
        LAC     CM          /SET ADDRESS IN WRITE STMT
        DAC     BOUT+2
        .ENTER  5,FILE
CLOCK   .TIMER  100,ADERR,7
        .TIMER  0,SYNC,6
        .IDLE
SYNC    0                   /RESTART ENTRY
        LAW     -120
        DAC     OUTSIZ#
        LAC     SEQ
        SAD     (50         /FIRST DISK FULL?
        JMP     CLOP
        SAD     (120        /SECOND DISK FULL?
        JMP     CLOP
        SAD     (160        /ALL DISKS FULL?
        JMP     CLOQ
LEFT    JMS ADCSET
        JMP .+3
        .DSA ONE            /ONE SAMPLE
        .DSA RETURN
        .IDLE
SAMP    0
ONE     1
TIMERR  0
N       1750                /1000(10) INPUT BUFFER SIZE
STOP    0
RETURN  RETURN+700001
        0                   /ENTRY (LEVEL 7)
        LAC BUFF1
        AND (740000         /IGNORE DATA
        DAC SAMP            /SAVE CHANNEL ID
        LAW -1              /INITIALIZE FOR COHINT
        DAC FREE1#
        LAC     CM
        DAC     OUTP#       /INCREMENTS BY 50 EACH INBUF
        DZM NDX#
        LAW -240
        DAC STOP            /ENABLE AUTOMATIC SWBUFF
        SPA!IAC             /WASTE 250US
        JMP .-1
        JMS ADCSET          /RESTART ADC DATA TAKING
        JMP .+3
        .DSA N
        .DSA PRADD          /ADDRESS OF COHERENT INTEGRATION ROUTINE
        .RLXIT  RETURN+1
CLOP    .CLOSE  5           /CLOSE FILE
        ISZ     EXT         /CHANGE NAME
        ISZ     CLOP        /INCREMENT DAT SLOT
        ISZ     BOUT        /WRITE ON NEXT DISK
        ISZ     .+1         /OPEN NEXT DISK
        .ENTER  5,FILE
        JMP     LEFT
CLOQ    .CLOSE  7           /ALL DISKS FULL QUIT
        .EXIT
PRADD   COHINT+500000
ADERR   0                   /ADC FAILURE (CLOCK EXPIRED)
        LAW     5774        /ISSUE TERMINAL ERROR
        DAC     ERCODE
        LAC     ADERR
        DAC     ERARG
        LAC*    (202
        ISA                 /PROTECT THE MONITOR
        JMS     ERROR
        DBK
        .RLXIT  ADERR
        .DEFIN  .INT M
```

Table A.1 cont'd.


```
        LAC     BUFF1,X+M
        LLS     12
        LRSS    12
        TAD     TEMP
        DAC     TEMP
        .ENDM
COHINT  0                                   /ENTRY
        LAC     TIMERR
        SNA
        JMP     TTT       /NO ERROR
        LAC     COHINT
        DAC     ERARG
        LAW     5773
        DAC     ERCODE
        LAC*    (202
        ISA
        JMS     ERROR
        DBK
        JMP     ENDIT
TTT     LAC     NDX
        PAX
        AAC     50
        PAL
        LAC     NDX       /SUBTRACT INDEX FROM ADDRESS
        TCA
        TAD     OUTP      /REMOVES INDEX OFFSET
        DAC     OUTX#
BEGINT  DZM     TEMP#
        .INT    0
        .INT    50
        .INT    120
        .INT    170
        .INT    240
        .INT    310
        .INT    360
        .INT    430
        .INT    500
        .INT    550
        .INT    620
        .INT    670
        .INT    740
        .INT    1010
        .INT    1060
        .INT    1130
        .INT    1200
        .INT    1250
        .INT    1320
        .INT    1370
        .INT    1440
        .INT    1510
        .INT    1560
        .INT    1630
        .INT    1700
        DAC*    OUTX,X
        AXS     1
        JMP     BEGINT
        LAW     -1
        DAC     COMFLG            /RELEASE INPUT BUFFER
        LAC     NDX               /SWITCH INPUT BUFFERS
        XOR     N
        DAC     NDX
        LAC     OUTP
        TAD     (50               /POINT TO NEXT TIME
        DAC     OUTP
        ISZ     OUTSIZ
        JMP     ENDIT
        ISZ     SEQ               /OUTPUT BUFFER FULL
        LAC     SEQ               /GET NEW SEQUENCE NUMBER
        DAC*    OUTX,X            /PUT IN OUTPUT BUFFER
```

Table A.1 cont'd.

```
        AXR     1
        LAC     SAMP            /CHID TO OUTBUF
        DAC*    OUTX,X
        AXR     1
        LAS
        DAC*    OUTX,X          /DATA SWITCHES TO OUTBUF
        ISZ     FREE1           /IS FIRST BUFFER FREE?
        JMP     SLOBO           /NO
BOUT    .REALW  5,4,CM,3203,SYNC,6
ENDIT   .RLXIT  COMINT
SLOBO   LAC     COMINT
        DAC     ERARG
        LAC*    (202    /PROTECT MONITOR
        ISA
        LAW     5776
        DAC     ERCODE
        JMS     ERROR
        DBK
        JMP     ENDIT
FILE    .SIXBT  '16MAY'
EXT     .SIXBT  '80A'
ADCSET  0               /ENTRY
        JMS*    .DA
        JMP     PAST
WC      .DSA    WC      /ADDRESS OF WORD COUNT
SUBR    .DSA    SUBR    /ADDRESS OF 'SUBR' ARG.
PAST    LAC     (404000 /RAISE TO PRIORITY LEVEL 4
        ISA
        LAC*    (155    /SETUP ADC ;ONCE ONLY
REAL    DAC     .
COMFLG  JMS*    .-1
DIFFC   703701
BPFLG   ADSVC
        DBK
SVAC    LAC*    (151
ERADD   DAC     REAL
        LAC     (JMP    IN
        DAC     PAST
IN      LAW     -1
        DAC     COMFLG
        TAD     (BUFF1
        DAC     BUF1#
        LAW     -1
        TAD     (BUFF2
        DAC     BUF2#
        TCA
        TAD     BUF1
        DAC     DIFFC
        DAC     BPFLG
        LAC*    WC
        TCA
        DAC     WC
        LAC*    (202
        ISA
        JMS     ADIN
        DBK
        JMP*    ADCSET  /RETURN
ADIN    0               /ENTRYY
        LAC     CLOCK+3 /RESET CLOCK
        DAC*    (7
        LAC     WC      /SETUP DCH
        DAC*    (26
        LAC     BPFLG
        TAD     BUF2
        DAC*    (27
        703704
        703744          /CLEAR FLAGS
        703724          /ENABLE TRANSFERS
        JMP*    ADIN    /RETURN
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.1 cont'd.

```
ADSVC   0                 /LEVEL 0 ENTRY
        DAC     SVAC      /SAVE AC
        703704            /CLEAR OFLO
        LAC     OUTSIZ    /TEST IF THIS IS LAST
        IAC
        SMA               /BUFFER FOR OUTPUT(WASTE NEXT ONE)
        DZM     STOP
        703721            /TIMING ERROR??
        JMP     .+3       /NO
        ISZ     TIMERR    /INFORM USER
        703744            /CLAAR FLAG
        ISZ     COMFLG    /SET BUSY FLG ZERO
        JMP     SLOWP     /IT ALREADY WAS--ERROR
        LAC*    SUBR
        SZA
        JMS*    REAL      /PRIME TO RUN SUBR AFTER EXIT
        LAC     DIFFC     /SWITCH BUFFERS
        XOR     BPFLG
        DAC     BPFLG
        LAC     STOP
        SZA
        JMS     ADIN
EXIT    LAC     (404000 /REQUEST MONITOR AFTER EXIT
        ISA
        LAC     SVAC
        DBR
        JMP*    ADSVC
SLOWP   LAC     ADSVC
        DAC     ERARG
        LAW     5777
        DAC     ERCODE
        JMS     ERROR
        JMP     EXIT
ERROR   0                 /ENTRY TO ERROR PRINTOUT
        LAC*    (166      /MONITOR ERROR SUBROUTINE
        DAC     ERADD
ERCODE  LAW     5770
        JMS*    ERADD
ERARG   777777            /ARGUMENT
        DZM     STOP
        JMP*    ERROR     /RETURN
BUFF1   .BLOCK  1750
BUFF2   .BLOCK  1750
        .END
```

Table A.2 CM.

```
        .GLOBL  CM
CM      .BLOCK  6203
        .END
```

in 20 minutes rather than 2 hours. Data collection for any length of time requires frequent unloading of disk to DECtape. The disks are nulled before starting and then assigned to slots 5, 6, and 7. If only one disk is available then it is assigned to all 3 of these slots. In either case the program SCAT and CM in object form should be available on a DECtape or disk assigned to slot -4. SCAT is the main program and so it must appear first in the loader command. After startup the program runs until all disks are full. These programs were written by Dr. Ian Countryman.

A.3 Power Printout Programs

The scattered power for the coherently integrated data is calculated and printed in tabular form on the PDP-15. A dc average for a given data file is calculated and the the file is read a second time to determine the power. The power is summed for one second or ten seconds depending on the program and printed for all 20 heights along with a time calculated from operator input. The programs exit when they have processed all records in the file.

The program CINTPW shown in Table A.3 is a PDP-15 FORTRAN main program for the calculation of power at 10 second intervals. It calls subroutines from a MACRO subprogram FTREAD which also must be loaded. The disk containing the data to be analyzed is assigned to slot 6, the foreground printing terminal to slot 2 and the program tape or disk with object code for these two programs to slot -4. The CINTPW program name appears before FTREAD in the loader input response because it is the main program.

The first section of the program requests the file name of the data, the start time of the first tape in a series of consecutively collected data tapes, the number of the tapes in the consecutive series, and the start setting of the radar director. In general, the coherently integrated data

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.3 CINTPW.

```
C  THIS PROGRAM CALCULATES 10 SECOND POWER VALUES FROM CINT DATA
C  COPY DATA FROM DECTAPE TO DISK USING PIP-IMAGE BINARY
C  ASSIGN TT TO SLOT 2
C  ASSIGN DISK WITH DATA TO SLOT 6
C  USES EXTERNAL MACOR ROUTINE FTREAD
C  DATA COLLECTED WITH SCAT AND CM PROGRAMS
C  KEN GIBBS  ?/2/82
         REAL FILEN(2),CO(20),CI(20),P(20)
         INTEGER IN(3203),OUT(20)
         LOGICAL OK
         COMMON /A/IN
C  WAKE UP THE DECWRITER
         WRITE(2,66)
66       FORMAT(1X,2HH )
C  GET FILE NAME AND CHECK FOR MATCH
70       WRITE(2,67)
67       FORMAT(1X,30HINPUT FILE NAME E.G. 14MAY 80A)
         READ(2,68) FILEN(1),FILEN(2)
68       FORMAT(A5,A4)
         CALL FSTAT(6,FILEN,OK)
         IF (OK) GO TO 84
         WRITE(2,69)
69       FORMAT (1X,24HMISMATCH...PLEASE RETYPE)
         GO TO 70
74       FORMAT(I2)
C  NOW CALCULATE A START TIME BASED ON WHICH TAPE IN A SERIES
C  THIS DATA IS AND THE START TIME OF FIRST TAPE
84       WRITE(2,85)
85       FORMAT(1X,43HIS THIS THE 1ST, 2ND, OR 3RD TAPE..2 DIGITS)
         READ(2,74) ITAPE
         IF (ITAPE.LT.1.OR.ITAPE.GT.4) GO TO 84
86       WRITE(2,87)
87       FORMAT(1X,42HINPUT START TIME OF FIRST TAPE IN THIS SET)
88       WRITE(2,89)
89       FORMAT(1X,23HHOURS-FROM 00 TO 23-CST)
         READ(2,74) IHOUR
         IF (IHOUR.LT.0.OR.IHOUR.GT.23) GO TO 89
90       WRITE(2,91)
91       FORMAT(1X,21HMINUTES-FROM 00 TO 59)
         READ(2,74)MINUT
         IF (MINUT.LT.0.OR.MINUT.GT.59) GO TO 91
C  CALCULATION OF TIME BASED ON THE ABOVE
         ISCND=400*(ITAPE-1)
         MINUT=MINUT+ISCND/60
         IF (MINUT.LT.59) GO TO 92
         MINUT=MINUT-60
         IHOUR=IHOUR+1
92       ISCND=ISCND-60*(ISCND/60)
C  NOW THE START HEIGHT INFO
94       WRITE (2,93)
93       FORMAT(1X,28HINPUT START SETTING-2 DIGITS)
         READ(2,74) ISS
         IF (ISS.LT.30.OR.ISS.GT.60) GO TO 94
C  SET UP READ ROUTINE WITH ADDRESS OF IN ARRAY
         CALL SET(IN)
C  FIND DC AVERAGES
         CALL SEEK(6,FILEN)
         DO 105 I=1,20
         CO(I)=0.0
105      CI(I)=0.0
         DO 79 I=1,40
         CALL GRAB
         IF (IN(3202).EQ.0) IROFF=20
         IF (IN(3202).NE.0) IROFF=0
         IIMOF=20-IROFF
         DO 79 JOFF=1,80
         DO 80 IHGT=1,20
         IROFF=IROFF+1
         IIMOF=IIMOF+1
```

Table A.3 cont'd.

ORIGINAL PAGE IS
OF POOR QUALITY

```
      CO(IHGT)=CO(IHGT)+FLOAT(IN(IROFF))
80    CI(IHGT)=CI(IHGT)+FLOAT(IN(IIMOF))
      IIMOF=IIMOF+20
79    IROFF=IROFF+20
      CALL CLOSE(6)
      DO 106 I=1,20
      CO(I)=CO(I)/3200.
106   CI(I)=CI(I)/3200.
C  PRINT BASE HEIGHT
C  ONLY GOOD FOR DATA  =1980
      BHGT=FLOAT(ISS-1)*1.5
      WRITE(2,605) BHGT
605   FORMAT(1X,9HBASE HGT=,F5.1,1X,2HKM)
      WRITE(2,650)
650   FORMAT(1X,35HPOWER IN TENTHS OF A DB ABOVE 60 DB)
      WRITE(2,615)
615   FORMAT(4X)
      WRITE(2,610) (IHGT,IHGT=1,20)
610   FORMAT(1X,4HTIME,3X,20I3)
      WRITE(2,615)
      CALL SEEK(6,FILEN)
C  LOOP OVER 10 SECOND RECORDS
      DO 202 I=1,40
      CALL GRAB
      IF (IN(3202).EQ.0)IROFF=20
      IF (IN(3202).NE.0) IROFF=0
      IIMOF=20-IROFF
      DO 205 IHGT=1,20
205   P(IHGT)=0.0
C  FIND THE SUM OF THE SQUARES OF RE AND IM
      DO 204 IOFF=1,80
      DO 203 IHGT=1,20
      IROFF=IROFF+1
      IIMOF=IIMOF+1
      AR=FLOAT(IN(IROFF))-CO(IHGT)
      AI=FLOAT(IN(IIMOF))-CI(IHGT)
203   P(IHGT)=P(IHGT)+AR*AR+AI*AI
      IIMOF=IIMOF+20
204   IROFF=IROFF+20
C TAKE LOG, MULT BY 100, SUBTRACT 600
C GIVES TENTHS OF DB ABOVE 60 DB
      DO 206 IHGT=1,20
206   OUT(IHGT)=IFIX(100.*ALOG10(P(IHGT)))-600
      WRITE(2,600) IHOUR,MINUT,ISCND, (OUT(J),J=1,20)
C  MAKE NEW TIME
      ISCND=ISCND+10
      IF (ISCND.LE.59) GOTO 202
      ISCND=ISCND-60
      MINUT=MINUT+1
      IF (MINUT.LE.59) GOTO 202
      MINUT=0
      IHOUR=IHOUR+1
202   CONTINUE
600   FORMAT(1X,3I2,1X,20I3)
C  DONE WITH DATA, INFORM USER, QUIT
      WRITE(2,550)
550   FORMAT(1X,19HDONE WITH THIS DATA)
      STOP
      END
```

are collected in a group of 3 tapes with the first tape starting on an integral minute. The second tape starts 400 seconds later, etc. The PDP-15 calculates the appropriate start time given the tape sequence number and the start time of the first tape in the sequence. The start setting is used by the computer to print the base altitude during output.

The next section of the program calculates the dc average for each altitude for the entire 400 seconds and saves the value to subtract during the power calculation. Note that the SET routine in the MACRO program must be called once to set up a buffer address in the GRAB routine. The SEEK command moves the file pointer to the start of the file and the GRAB command reads in a 10-second record each time it is called. The FTREAD program and its routines are discussed below.

The structure of the commands for separation of the sine and cosine channel outputs and the 20 heights is based on the record structure shown in Figure A.1. The input buffer is 3203 integer words of which the first 3200 form a table of 20 cosine channel plus 20 sine channel outputs every 1/8 second for 10 seconds. In order to determine which channel the left half of the table corresponds to (i.e., the first 20 elements of each group of 40) the channel indicator in location 3202 of the array must be examined. If this value is 0 then channel 0 is on the right. If element 3202 is nonzero then the opposite holds. Essentially the number in element 3202 is the opposite of the channel number of the first element in the array. In each half of the array the lowest altitude has the lowest index and the earliest time also has the lowest index. Element 3201 contains the sequence number which is the number of records in the sequence of tapes before the present one. If the present record is record 7 of tape 2 in a sequence then element 3201 will contain 46 because 40 records were written to the first tape and 6

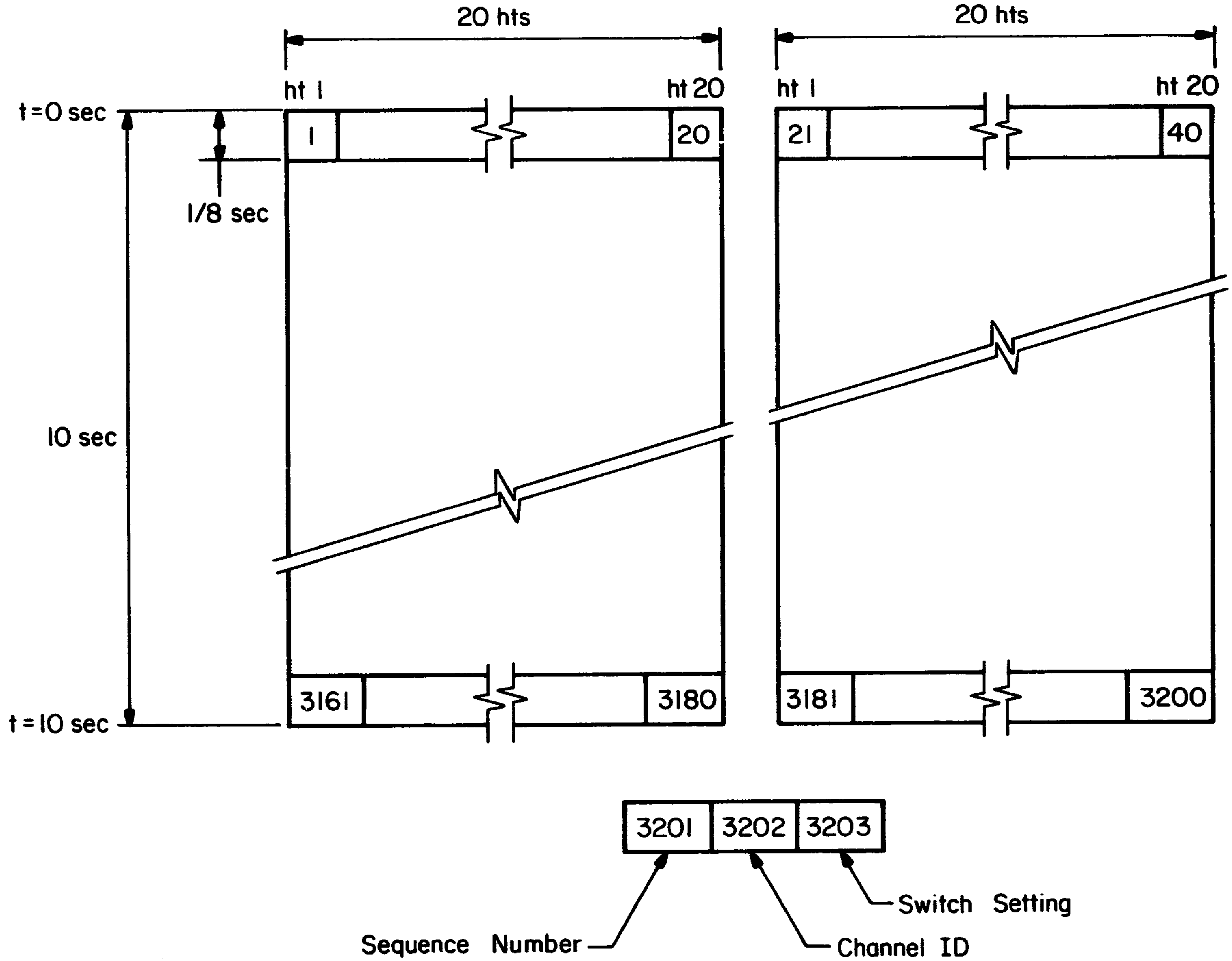


Figure A.1 One record of a coherently integrated data file collected by SCAT, CM programs.

previously were written on the present tape. Element 3203 conta : the setting of the PDP-15 front panel switches during data collection. The sequence number and switch settings are generally not used.

After calculating the dc offsets a table header is printed. Each record is read in and the power for each altitude is summed for the entire record. This value is converted to dB and 60 dB is subtracted. A line of the output table is printed with integers corresponding to tenths of a dB above 60 dB. The record time is also printed. The program halts when done with all 40 records in the file.

The CINTIS program shown in Table A.4 is used to print the power values at 1 second intervals. It is essentially the same as the CINTPW program but the conversion to dB and printing must occur 10 times for each record in the data file. Values are printed as tenths of a dB above 50 dB. The I/O slot assignments and operation of the program are identical with those of the CINTPW program. Like CINTPW, CINTIS is the main program and therefore its name must appear before FTREAD in the loader command.

The FTREAD program illustrated in Table A.5 is a MACRO program originally written to read the coherently integrated data files for the FFT programs which are discussed below. The power programs discussed above also use FTREAD to read the nonstandard length records which cannot be read directly from FORTRAN I/O library calls. The routine SET is called first from the accompanying FORTRAN program in the form CALL SET(IN) where IN is an integer array of length 3203 which has been declared in a COMMON block to allow transfer from the MACRO to FORTRAN program. The address of the IN array is passed to the SET routine which puts it into the read statement labeled R. A more detailed explanation of this process is beyond the scope of this discussion. The GRAB routine is called from FORTRAN to read a

Table A.4 CINT1S.

ORIGINAL PAGE IS OF POOR QUALITY

```
C THIS PROGRAM CALCULATES 1 SECOND POWER VLAUES FROM CINT DATA
C  COPY DATA FROM DECTAPE TO DISK USING PIP-IMAGE BINARY
C  ASSIGN TT TO SLOT 2
C  ASSIGN DISK WITH DATA TO SLOT 6
C  USES EXTERNAL MACRO ROUTINE FTREAD
C  DATA COLLECTED WITH SCAT AND CM PROGRAMS
C  KEN GIBBS  5/12/82
       REAL FILEN(2),CO(20),CI(20),P(20)
       INTEGER IN(3203),OUT(20)
       LOGICAL OK
       COMMON /A/IN
C  WAKE UP THE DECWRITER
       WRITE(2,66)
66     FORMAT(1X,2HH )
C  GET FILE NAME AND CHECK FOR MATCH
70     WRITE(2,67)
67     FORMAT(1X,30HINPUT FILE NAME E.G. 14MAY 80A)
       READ(2,68) FILEN(1),FILEN(2)
68     FORMAT(A5,A4)
       CALL FSTAT(6,FILEN,OK)
       IF (OK) GO TO 84
       WRITE(2,69)
69     FORMAT (1X,24HMISMATCH...PLEASE RETYPE)
       GO TO 70
74     FORMAT(I2)
C  NOW CALCULATE A START TIME BASED ON WHICH TAPE IN A SERIES
C  THIS DATA IS AND THE START TIME OF FIRST TAPE
84     WRITE(2,85)
85     FORMAT(1X,43HIS THIS THE 1ST, 2ND, OR 3RD TAPE..2 DIGITS)
       READ(2,74) ITAPE
       IF (ITAPE.LT.1.OR.ITAPE.GT.4) GO TO 84
86     WRITE(2,87)
87     FORMAT(1X,42HINPUT START TIME OF FIRST TAPE IN THIS SET)
88     WRITE(2,89)
89     FORMAT(1X,23HHOURS-FROM 00 TO 23-CST)
       READ(2,74) IHOUR
       IF (IHOUR.LT.0.OR.IHOUR.GT.23) GO TO 89
90     WRITE(2,91)
91     FORMAT(1X,21HMINUTES-FROM 00 TO 59)
       READ(2,74)MINUT
       IF (MINUT.LT.0.OR.MINUT.GT.59) GO TO 91
C  CALCULATION OF TIME BASED ON THE ABOVE
       ISCND=400*(ITAPE-1)
       MINUT=MINUT+ISCND/60
       IF (MINUT.LT.59) GO TO 92
       MINUT=MINUT-60
       IHOUR=IHOUR+1
92     ISCND=ISCND-60*(ISCND/60)
C  NOW THE START HEIGHT INFO
94     WRITE (2,93)
93     FORMAT(1X,28HINPUT START SETTING-2 DIGITS)
       READ(2,74) ISS
       IF (ISS.LT.30.OR.ISS.GT.60) GO TO 94
C  SET UP READ ROUTINE WITH ADDRESS OF IN ARRAY
       CALL SET(IN)
C  FIND DC AVERAGES
       CALL SEEK(6,FILEN)
       DO 105 I=1,20
       CO(I)=0.0
105    CI(I)=0.0
       DO 79 I=1,40
       CALL GRAB
       IF (IN(3202).EQ.0) IROFF=20
       IF (IN(3202).NE.0) IROFF=0
       IIMOF=20-IROFF
       DO 79 JOFF=1,80
       DO 80 IHGT=1,20
       IROFF=IROFF+1
       IIMOF=IIMOF+1
```

Table A.4 cont'd.

```
      CO(IHGT)=CO(IHGT)+FLOAT(IN(IROFF))
80    CI(IHGT)=CI(IHGT)+FLOAT(IN(IIMOF))
      IIMOF=IIMOF+20
79    IROFF=IROFF+20
      CALL CLOSE(6)
      DO 106 I=1,20
      CO(I)=CO(I)/3200.
106   CI(I)=CI(I)/3200.
C  PRINT BASE HEIGHT
C  ONLY GOOD FOR DATA >=1980
      BHGT=FLOAT(ISS-1)*1.5
      WRITE(2,605) BHGT
605   FORMAT(1X,9HBASE HGT=,F5.1,1X,2HKM)
      WRITE(2,650)
650   FORMAT(1X,35HPOWER IN TENTHS OF A DB ABOVE 50 DB)
      WRITE(2,615)
615   FORMAT(4X)
      WRITE(2,610) (IHGT,IHGT=1,20)
610   FORMAT(1X,4HTIME,3X,20I3)
      WRITE(2,615)
      CALL SEEK(6,FILEN)
C  THERE ARE 40 RECORDS OF 10 SECONDS DATA
      DO 202 I=1,40
      CALL GRAB
      IF (IN(3202).EQ.0)IROFF=20
      IF (IN(3202).NE.0) IROFF=0
      IIMOF=20-IROFF
C  LOOP OVER 10 SECONDS PER RECORD
      DO 202 ISC=1,10
C  INIT P ARRAY FOR RUNNING SUM
      DO 205 IHGT=1,20
205   P(IHGT)=0.0
C  8 SAMPLES PER SECOND
      DO 204 ISAMP=1,8
      DO 203 IHGT=1,20
      IROFF=IROFF+1
      IIMOF=IIMOF+1
      AR=FLOAT(IN(IROFF))-CO(IHGT)
      AI=FLOAT(IN(IIMOF))-CI(IHGT)
203   P(IHGT)=P(IHGT)+AR*AR+AI*AI
C  SHIFT POINTERS TO NEXT SAMPLE
      IIMOF=IIMOF+20
204   IROFF=IROFF+20
C TAKE LOG, MULT BY 100, SUBTRACT 500
C GIVES TENTHS OF DB ABOVE 50 DB
      DO 206 IHGT=1,20
206   OUT(IHGT)=IFIX(100.*ALOG10(P(IHGT)))-500
      WRITE(2,600) IHOUR,MINUT,ISCND, (OUT(J),J=1,20)
C  MAKE NEW TIME
      ISCND=ISCND+1
      IF (ISCND.LE.59) GOTO 202
      ISCND=ISCND-60
      MINUT=MINUT+1
      IF (MINUT.LE.59) GOTO 202
      MINUT=0
      IHOUR=IHOUR+1
202   CONTINUE
600   FORMAT(1X,3I2,1X,20I3)
C  DONE WITH DATA, INFORM USER, QUIT
      WRITE(2,550)
550   FORMAT(1X,19HDONE WITH THIS DATA)
      STOP
      END
```

Table A.5 FTREAD.


```
/  THIS ROUTINE IS USED WITH THE FFT CALCULATION MAIN PROGRAM
/  BECAUSE FORTRAN IS UNABLE TO READ A BUFFER OF 3203 WORDS.
/  THE ROUTINE SET IS CALLED TO PUT THE ADDRESS OF THE BUFFER
/  INTO THE READ STATEMENT IN THE SECOND ROUTINE.
/  THE ROUTINE GRAB DOES THE ACTUAL READ OF THE DATA WHEN CALLED.
/  THE ROUTINE SWITCH LOOKS FOR A 0 TO 1 TRANSITION
/  ON THE BIT 0 SWITCH BEFORE ALLOWING FFT TO BE SENT
/  KEN GIBBS 2/26/82
        .GLOBL   SET,GRAB,SWITCH,.DA
        .IODEV   6
SET     0
        JMS*     .DA
        JMP      .+2
ARR     0
        LAC*     ARR
        DAC      R+2
        JMP*     SET
GRAB    0
R       .READ 6,4,ARR,3203
        .WAIT 6
        JMP*     GRAB
SWITCH  0
LOOK0   CLA!CLL /BE SURE BIT 0 GOES TO ZERO
        OAS
        RAL
        SZL
        JMP      LOOK0
LOOK1   CLA!CLL /NOW LOOK FOR THE 0 TO 1 TRANSITION
        OAS
        RAL
        SNL
        JMP      LOOK1
        JMP*     SWITCH
        .END
```

single record from the file. All other handling of I/O is done via calls to the FORTRAN I/O library.

The SWITCH routine is called from FORTRAN to cause the PDP-15 to suspend processing until a 0 to 1 transition occurs on the bit 0 switch of the front panel. The PDP-15 is put into the LOOKO loop waiting for a zero. When it finds the switch set to zero it moves to the LOOK1 loop waiting for a one. When the 1 is found control returns to the FORTRAN calling program and processing resumes. The SWITCH routine is used by the FFT transfer programs to control processing without requiring any communication with the terminal. In this way no extra characters which might interfere with the PDP-15 to Apple II transfer process are sent. The use of the FFT programs and the transfer to Apple II disk are discussed below.

A.4 Processing and Transfer of 32-, 64-, and 128-Second Spectra

There are two groups of programs for the calculation and transfer of power spectra from the coherently integrated data. The first set is used for 32-, 64-, and 128-second spectra. These spectra are calculated at 30-, 60-, and 120-second intervals, respectively, which requires a small overlap between spectra and complicates the processing programs. Power spectra of length 8 seconds are calculated at 10 second intervals using a second set of programs. Processing of the data to obtain spectra involves the calculation of the spectra on the PDP-15 with essentially concurrent transfer and storage to Apple II floppy disk and a second step in which the power spectra are transferred from the Apple II to the HP-9830 for plotting. These transfer and plotting programs are similar to the procedures given by Roth [1982] for the routine coherent-scatter data. The primary differences are in the calculation routines and the file structure on the Apple II floppy disks.

The PDP-15 FORTRAN main program for the calculation and transfer of 32-, 64-, and 128-second power spectra, called FFTAMN, is shown in Table A.6. The majority of this program is similar to the CINTPW program because the same PDP-15 data files are used. In addition to the input required by the CINTPW programs, FFTAMN also requires the interval between spectra. After input is complete various parameters for control of the FFT routine and the coefficients for the FFTs are calculated followed by calculation of the dc offset of the data. At that point the transfer to the Apple II begins.

It should be noted that the PDP-15 is unaware of the transfer to the Apple II -- it simply writes a special string of values preceded by several asterisks to a terminal. The Apple II can be configured to read this string of characters on the PDP-15 terminal as described in Roth [1982]. The PDP-15 can calculate data faster than the Apple II can read, interpret, and write it to disk. Since the amount of data to transfer exceeds the memory capacity of the Apple II the PDP-15 must be interrupted. Furthermore, since the Apple II is monitoring the PDP-15 terminal the interruption is generated at the PDP-15 switch panel. The purpose of the SWITCH routine in the FTREAD MACRO program above is to create and handle this interrupt. The operator toggles the bit 0 switch when the Apple II indicates that it ready for more data.

After calculating the dc offset the PDP-15 immediately sends a header to the Apple II which must have its program already running. While the Apple II is interpreting the header, checking for floppy disks, etc., the PDP-15 begins calculation of the first FFT. A set of temporary arrays must be used for the coherently integrated data to save overlap data which would otherwise be written over by the in-place FFT algorithm. The alternative is

Table A.6 FFTAMN.


```
C  THIS PROGRAM CALCULATES FFTS OF COHERENTLY INTEGRATED DATA
C  COPY DATA FROM DECTAPE TO DISK USING PIP-IMAGE BINARY
C  ASSIGN TT TO SLOT 2, USE FCONTROL1 AND FT1 FOR ACTUAL TRANSFER
C  ASSIGN DISK WITH DATA TO SLOT 6
C  DATA COLLECTED WITH SCAT AND CM PROGRAMS
C  USES EXTERNAL MACRO ROUTINE FTREAD
C  USES EXTERNAL FORTRAN ROUTINES FTCALC AND FTFILL
C  KEN GIBBS  2/25/82
       REAL RE(1024),AIM(1024),COR(512),COI(512),FILEN(2)
       INTEGER IN(3203),HR(1040),HI(1040),DR(20),DI(20),OUT(1024)
       LOGICAL OK
       DIMENSION CO(20),CI(20)
       COMMON /A/IN,HR,HI/F/N,M,NV2,NM1,RE,AIM,COR,COI
C  WAKE UP THE DECWRITER
       WRITE(2,66)
66     FORMAT(1X,2HH )
C  GET FILE NAME AND CHECK FOR MATCH
70     WRITE(2,67)
67     FORMAT(1X,30HINPUT FILE NAME E.G. 14MAY 80A)
       READ(2,68) FILEN(1),FILEN(2)
68     FORMAT(A5,A4)
       CALL FSTAT(6,FILEN,OK)
       IF (OK) GO TO 84
       WRITE(2,69)
69     FORMAT (1X,24HMISMATCH...PLEASE RETYPE)
       GO TO 70
74     FORMAT(I2)
C  NOW CALCULATE A START TIME BASED ON WHICH TAPE IN A SERIES
C  THIS DATA IS AND THE START TIME OF FIRST TAPE
84     WRITE(2,85)
85     FORMAT(1X,43HIS THIS THE 1ST, 2ND, OR 3RD TAPE..2 DIGITS)
       READ(2,74) ITAPE
       IF (ITAPE.LT.1.OR.ITAPE.GT.4) GO TO 84
86     WRITE(2,87)
87     FORMAT(1X,42HINPUT START TIME OF FIRST TAPE IN THIS SET)
88     WRITE(2,89)
89     FORMAT(1X,23HHOURS-FROM 00 TO 23-CST)
       READ(2,74) IHOUR
       IF (IHOUR.LT.0.OR.IHOUR.GT.23) GO TO 89
90     WRITE(2,91)
91     FORMAT(1X,21HMINUTES-FROM 00 TO 59)
       READ(2,74)MINUT
       IF (MINUT.LT.0.OR.MINUT.GT.59) GO TO 91
C  CALCULATION OF TIME BASED ON THE ABOVE
       ISCND=400*(ITAPE-1)
       MINUT=MINUT+ISCND/60
       IF (MINUT.LT.59) GO TO 92
       MINUT=MINUT-60
       IHOUR=IHOUR+1
92     ISCND=ISCND-60*(ISCND/60)
C  NOW THE START HEIGHT INFO
94     WRITE (2,93)
93     FORMAT(1X,28HINPUT START SETTING-2 DIGITS)
       READ(2,74) ISS
       IF (ISS.LT.30.OR.ISS.GT.60) GO TO 94
C  SET UP PARAMETERS FOR FFT
95     WRITE(2,72)
72     FORMAT(1X,47HINPUT INTERVAL BETWEEN FFTS-30, 60, OR 120 SECS)
       READ(2,200)ISEC
200    FORMAT(I5)
       IF (ISEC.LE.30) ISEC=30
       IF ((ISEC.GT.30).AND.(ISEC.LE.60)) ISEC=60
       IF (ISEC.GT.60) ISEC=120
       IRECM=ISEC/10+1
       M=ISEC/30+7
       IF (M.EQ.11) M=10
       LFILL=(IRECM-1)*80+1
       N=2**M
       NV2=N/2
```

Table A.6 cont'd.


```
      NM1=N-1
      IHTMX=20
      IEND=3072/N
      IHTMN=1
      ISTRT=1
      NV16=N/16
      ZERO=0.0
C  FILL COEFFECIENT ARRAYS
      PI=3.14159265
      F1=PI/FLOAT(NV2)
      ANGLE=0.0
      DO 400 I=1,NV2
      COR(I)=COS(ANGLE)
      COI(I)=-SIN(ANGLE)
400   ANGLE=ANGLE+F1
C  SET UP READ ROUTINE WITH ADDRESS OF IN ARRAY
      CALL SET(IN)
C  FIND DC AVERAGES
      CALL SEEK(6,FILEN)
      DO 105 I=1,20
      CO(I)=0.0
105   CI(I)=0.0
      DO [illegible] I=1,40
      CALL GRAB
      IF (IN(3202).EQ.0) IROFF=20
      IF (IN(3202).NE.0) IROFF=0
      IIMOF=20-IROFF
      DO 79 JOFF=1,80
      DO 80 IHGT=1,20
      IROFF=IROFF+1
      IIMOF=IIMOF+1
      CO(IHGT)=CO(IHGT)+FLOAT(IN(IROFF))
80    CI(IHGT)=CI(IHGT)+FLOAT(IN(IIMOF))
      IIMOF=IIMOF+20
79    IROFF=IROFF+20
      CALL CLOSE(6)
      DO 106 I=1,20
      DR(I)=IFIX(CO(I)/3200.)
106   DI(I)=IFIX(CI(I)/3200.)
C  SEND THE ONE TIME ONLY STUFF TO THE APPLE
      WRITE(2,300) FILEN(1),FILEN(2),IHOUR,MINUT,ISCND,ISS,N
300   FORMAT(1X,8H********,2A5,3I8,2I7,1H*)
      DO 302 J=1,16,5
      L2=J+4
      WRITE(2,301) (DR(I),I=J,L2)
302   WRITE(2,301) (DI(I),I=J,L2)
301   FORMAT(1X,8H********,5I8,9H*********)
C  INFORM USER OF SWITCH STUFF
      WRITE(2,303)
303   FORMAT(1X,51HPDP-15 WILL WAIT FOR SIGNAL TO BEGIN SENDING AN FFT)
      WRITE(2,304)
304   FORMAT(1X,45HA 0 TO 1 TRANSITION ON THE BIT 0 SWITCH=START)
C  BEGIN PROCESSING, LOOP OVER HEIGHT
      DO 55 IHGT=IHTMN,IHTMX
      CALL SEEK(6,FILEN)
      IFILL=1
C  IFNO=NO. OF FFT PER HEIGHT
      DO 50 IFNO=ISTRT,IEND
      IF (IFNO.GT.1) GO TO 52
      CALL FILLER(IHGT,IFILL)
      GO TO 53
C  WHEN NOT THE FIRST FFT AT A HEIGHT MOVE LAST
C  10 SECONDS TO FIRST 10 SECONDS OF HR,HI ARRAYS
C  THEN CONTINUE TO FILL
52    IFILL=LFILL
      DO 37 JOFF=1,80
      HI(JOFF)=HI(IFILL)
      HR(JOFF)=HR(IFILL)
37    IFILL=IFILL+1
```

Table A.6 cont'd.

```
        IFILL=81
53      DO 51 IREC=2,IRECM
51      CALL FILLER(IHGT,IFILL)
C  TRANSFER TO RE, AIM AND SUBTRACT DC
        DO 54 I=1,N
        RE(I)=FLOAT(HR(I)-DR(IHGT))
54      AIM(I)=FLOAT(HI(I)-DI(IHGT))
C  TRANSFORM
        CALL FFT
C  FIND POWER SPECTRUM,NORMALIZE,FIX
        HIGH=RE(1)
        DO 600 I=1,N
        RE(I)=RE(I)*RE(I)+AIM(I)*AIM(I)
600     IF (HIGH.LT.RE(I)) HIGH=RE(I)
        FUDGE=999.0/HIGH
        DO 601 I=1,N
601     OUT(I)=IFIX(RE(I)*FUDGE)
C  WAIT FOR APPLE OK SIGNAL
        CALL SWITCH
C  SEND HGTNO,FFTNO,FUDGE FACTOR
        WRITE(2,602)IHGT,IFNO,FUDGE,ZERO,ZERO
602     FORMAT(1X,8H********,2I8,E12.5,2F10.1,1H*)
C  SEND FFT DATA
        DO 605 I=1,NV16
        L1=(I-1)*16+1
        L2=L1+15
605     WRITE(2,606) (OUT(J),J=L1,L2)
606     FORMAT(1X,8H********,16I3,1H*)
50      CONTINUE
55      CALL CLOSE(6)
C  DONE WITH DATA, INFORM USER, QUIT
        WRITE(2,550)
550     FORMAT(1X,19HDONE WITH THIS DATA)
        STOP
        END
```

a time consuming scan through the data file to reposition the file pointer at the previous record of the file before the next FFT can be calculated. Using a temporary array allows each record of the data file to be read only once for each height. Ideally this program would operate on all 20 heights at once, reading each record from disk one time. However, the temporary array would then be 20 times as large, exceeding the storage capacity of the PDP-15.

The FFT arrays are filled from the temporary arrays and the dc offset is subtracted. (The temporary arrays might hold 70 seconds of data of which 64 seconds will be used for the FFT.) The FFT is performed by the routine FFT in the program FTCALC which is discussed below. The power spectrum is calculated from the sum of the squares of the real and imaginary parts of the transform. The output array is the power spectrum normalized so that the maximum value is 999 and the values can be output as integers. After calculation the PDP-15 waits on the switch routine before sending the results to the Apple II. While the Apple II stores the power spectrum the PDP-15 moves the last 10 seconds of the temporary arrays to the first 10 seconds, fills the remainder of the temporary arrays and then calculates the next FFT at that height. The process continues until all FFTs at one height have been transferred. The file pointer in the PDP-15 data file is returned to the beginning of the file and the process is repeated for the next height until all heights have been processed.

The FFTAMN program as shown here must process all spectra for a given data file. It has been written, however, so that by changing some of the statements from internally set constants to user input variables the number of FFTs can be reduced. Specifically, the variables IHTMN and IHTMX determine the range of heights which are processed. If these are input instead

of set the number of heights can be reduced.

The FTFILL program shown in Table A.7 is used to move records from the data disk to the temporary arrays used by the FFTAMN program. The use of the GRAB routine and the statements to sort through the input buffer are similar to the statements used to determine the dc offset. The routine is called with the command CALL FTFILL (IHGT, IFILL) where IHGT is an integer which indicates the present height being processed and IFILL is a pointer to the next location in the temporary arrays which is to be filled. The routine exits with IFILL set to the next empty location so that FTFILL can be called repeatedly.

The FTCALC program shown in Table A.8 is used to perform the FFT calculations. It is called without parameters as CALL FFT because the required arrays and constants are passed via a COMMON declaration. The definitions of these variables and constants are given in Table A.9. The FFT routine is a power of two algorithm derived from a routine given by Oppenheim and Schafer [1975]. The first two stages of the calculation have been separated to reduce the number of multiplications per FFT. As mentioned above the coefficients are calculated externally to further increase speed. The routine is an in-place algorithm so the FFT is returned in the input array. The FTCALC routine in conjunction with the FFTAMN program will do the array transfers, subtract dc, calculate the FFT, and determine the normalized power spectrum for a 512 point FFT in under 6 seconds.

The FFTAMN is the main program used with the FTREAD, FTCALC, and FTFILL programs and must be specified first in the loader command. The object version of these programs must be on a device assigned to slot -4. The data disk is assigned to slot 6 as in the power print program. The transfer of information to the Apple II requires that the terminal assignments be modi-

ORIGINAL PAGE IS OF POOR QUALITY

Table A.7 FTFILL.

```
C  THIS ROUTINE IS USED WITH THE FFT CALCULATION MAIN PROGRAM
C  IT IS USED TO PICK OFF THE APPROPRIATE HEIGHT AND STORE IT
C  IN THE SET UP ARRAYS HR AND HI
C  ENTER WITH IFILL POINTING TO FIRST LOCATION TO FILL
C  EXIT WITH IFILL POINTING TO NEXT LOCATION TO FILL
         SUBROUTINE FILLER(IHGT,IFILL)
         INTEGER IN(3203),HR(1040),HI(1040)
         COMMON /A/IN,HR,HI
         CALL GRAB
         IF (IN(3202).EQ.0) GO TO 33
         IROFF=IHGT
         IIMOF=IHGT+20
         GO TO 34
33       IIMOF=IHGT
         IROFF=IHGT+20
34       DO 31 JOFF=1,80
         HR(IFILL)=IN(IROFF)
         HI(IFILL)=IN(IIMOF)
         IIMOF=IIMOF+40
         IROFF=IROFF+40
31       IFILL=IFILL+1
         RETURN
         END
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.8 FTCALC.

```
C  THIS ROUTINE CALCULATES POWER OF TWO LENGTH FFTS
C  INPUT AND OUTPUT ARE PASSED VIA THE RE AND AIM ARRAYS
C  N IS THE NUMBER OF POINTS, NM1=N-1, NV2=N/2
C  COR AND COI ARE COEFFECIENT ARRAYS EXTERNALLY CALCULATED
C  THIS IS BASED ON A DECIMATION IN TIME ALGORITHM
C  THE FIRST AND SECOND STAGES HAVE BEEN SEPARATED
C  TO REDUCE COMPUTATION TIME
C  KEN GIBBS 11/24/81
         SUBROUTINE FFT
         REAL RE(1024),AIM(1024),COR(512),COI(512)
         COMMON /F/N,M,NV2,NM1,RE,AIM,COR,COI
C        SHUFFLE HERE
         J=1
         DO 7 I=1,NM1
         IF (I.GE.J) GO TO 5
         TR=RE(J)
         TI=AIM(J)
         RE(J)=RE(I)
         AIM(J)=AIM(I)
         RE(I)=TR
         AIM(I)=TI
5        K=NV2
6        IF (K.GE.J) GO TO 7
         J=J-K
         K=K/2
         GO TO 6
7        J=J+K
C        FIRST STAGF OF FFT, NO MULTIPLY
         DO 8 I=1,N,2
         IP=I+1
         TR=RE(IP)
         TI=AIM(IP)
         RE(IP)=RE(I)-TR
         AIM(IP)=AIM(I)-TI
         RE(I)=RE(I)+TR
8        AIM(I)=AIM(I)+TI
C        SECOND STAGE OF FFT, NO MULTIPLIES
         DO 9 J=1,2
         DO 9 I=J,N,4
         IP=I+2
         IF (J.EQ.2) GO TO 11
         TR=RE(IP)
         TI=AIM(IP)
11       IF (J.EQ.1) GO TO 12
         TR=AIM(IP)
         TI=-RE(IP)
12       RE(IP)=RE(I)-TR
         AIM(IP)=AIM(I)-TI
         RE(I)=RE(I)+TR
9        AIM(I)=AIM(I)+TI
C        STAGES 3 TO M OF FFT
         LE=4
         DO 10 L=3,M
         LE1=LE
         LE=LE*2
         NV3=NV2/LE1
         LOC=1
         DO 10 J=1,LE1
         UR=COR(LOC)
         UI=COI(LOC)
         LOC=LOC+NV3
         DO 10 I=J,N,LE
         IP=I+LE1
         TR=RE(IP)*UR-AIM(IP)*UI
         TI=RE(IP)*UI+AIM(IP)*UR
         RE(IP)=RE(I)-TR
         AIM(IP)=AIM(I)-TI
         RE(I)=RE(I)+TR
10       AIM(I)=AIM(I)+TI
         RETURN
         END
```

Table A.9 Variables and constants used by the FFT routine.

N: The number of points in the FFT. This must be a power of two and less than or equal to 1024.

$M = \log_2 N$

NV2 = N/2

NM1 = N-1

RE, AIM: These arrays contain the real and imaginary values that are the input and output to the FFT algorithm. RE(1) is the real component at zero frequency for example. The highest frequency is at element N of the arrays. Negative frequencies should be folded over to positive for this algorithm. In the time domain the earliest time is in element 1, the latest time in element N.

COR, COI: These arrays contain the real and imaginary parts of the multipliers for the FFT algorithm and are calculated externally to FTCALC (see FFTAMN). The elements represent consecutive values on the unit circle in the complex plane in negative increments of $2\pi/N$ radians. Element 1 corresponds to 0 times the increment while element NV2 corresponds to NV2-1 times the angular increment. Only the bottom half of the unit circle is needed.

fied. The transfer is designed to work with the video terminal on the PDP-15 at 9600 baud. The foreground job must be run on this terminal so an FCONTROL 1 monitor command should be used. The output terminal is then terminal 1 and the correct assignment statement for the terminal is TT1 to slot 2.

The program FFTTOAPDISKSRC shown in Table A.10 is written in Applesoft, the floating-point BASIC language for the Apple II. This program is used to transfer the 32-, 64-, or 128-second FFTs generated by the PDP-15 program shown above. The title of the program ends in SRC to indicate that it has been compiled with the TASC compiler and that the compiled version must be used. A small difference between the compiler and interpreter in the handling of strings makes the source code running under the interpreter behave differently than the object code generated by the compiler. The object program needed is called FFTTOAPDISK.OBJ and must be loaded with an interface driver routine AP15DRIVER. Whenever a TASC compiled program is used the RUNTIME library must also be loaded. The correct sequence is BLOAD RUNTIME, BLOAD AP15DRIVER, BRUN FFTTOAPDISK.OBJ. The AP15DRIVER will be discussed below.

FFTTOAPDISKSRC reads data from the PDP-15 and transfers the data to the Apple II floppy disks. These two processes occur at separate times because of the relative speeds and the lack of a handshake between the Apple II and the PDP-15. The program therefore consists of initialization in lines 100 to 120, reading the header information from the PDP-15 in lines 1000 to 1027, parsing the header information in lines 1030 to 1450, disk and error handling routines in lines 4200 to 7020, and a large loop between lines 1500 and 2310 which reads and stores the FFTs.

Table A.10 FFTTOAPDISKSRC.

ORIGINAL PAGE IS
OF POOR QUALITY

```
1  REM  TASC COMPILER SOURCE ONLY
2  REM  WILL NOT RUN WITH AP15DRIVER
3  REM  UNDER THE INTERPRETER
4  REM  KEN GIBBS 3/10/82
5  REM  ! INTEGER I,FM,TO,CT,J,N6,HN,NF
6  REM  ! INTEGER SL,DR,VL,ST
100 M$ = "1234567890123456789012345678901234567890123456789012345678901234567"
101 M$ = M$ + "8"
102 S$ = " ":D$ =  CHR$ (4)
103 SL = 6:DR = 1:VL = 0
105  DIM E$(12),A$(64),HT$(20),DR$(20),DI$(20),T$(13)
110  REM  SET UP MONTH TABLE
111  FOR I = 1 TO 12: READ E$(I): NEXT
112  DATA  "JANUARY","FEBRUARY","MARCH","APRIL","MAY"
113  DATA  "JUNE","JULY","AUGUST","SEPTEMBER"
114  DATA  "OCTOBER","NOVEMBER","DECEMBER"
120  PRINT : PRINT "WAITING FOR HEADER"
1000  REM  READ HEADER AND DC
1001  REM  GET FILENAME,HOURS,MINUTES,SECONDS,START SETTING,NO. OF POINTS

1005  CALL 832
1007 B$ = M$ + S$
1008  REM  GET DC
1010  FOR I = 1 TO 8
1015  CALL 832
1020 A$(I) = M$ + S$
1025  NEXT I
1027  PRINT : PRINT "GOT HEADER STUFF"
1030  REM  GET DATE FROM FILENAME
1031  REM  SCAN THIRD CHARACTER OF MONTH
1040  FOR I = 1 TO 12
1045  IF  MID$ (B$,5,1) =  MID$ (E$(I),3,1) THEN MN$ = E$(I): GOTO 1060
1050  NEXT
1055  PRINT "FILENAME ERROR": STOP
1060  REM  CHECK SECOND LETTER FOR SPECIAL CASES
1065  REM  LOOP ABOVE PICKS FIRST MATCH, SWITCH TO SECOND IF NECESSARY
1070  IF  MID$ (B$,5,1) = "N" AND  MID$ (B$,4,1) = "U" THEN MN$ = E$(6): GOTO
     1080
1075  IF  MID$ (B$,5,1) = "R" AND  MID$ (B$,4,1) = "P" THEN MN$ = E$(4)
1080 H$ = MN$ + S$ +  STR$ ( VAL ( LEFT$ (B$,2))) + "." + S$ + "19" +  MID$
     (B$,7,2)
1085  REM  H$=DATE WITH DOT AS STRING=HEADER FILE NAME
1100  REM  NOW MAKE UP TIME STRINGS
1105  REM  GET HOURS,MINUTES,SECONDS
1110 T4 =  VAL ( MID$ (B$,17,2))
1115 T3 =  VAL ( MID$ (B$,25,2))
1120 T1 =  VAL ( MID$ (B$,33,2))
1125  REM  FIND NO. OF SECONDS BETWEEN FFTS-FIRST GET N
1130 N3 =  VAL ( MID$ (B$,45,4))
1135  REM  T2=SECONDS BETWEEN FFTS
1140 T2 = N3 * 120 / 1024
1145  REM  FM=MAX. NO OF FFTS
1150 FM = 3072 / N3
1155  REM  NOW LOOP TO SET UP THE TIME STRINGS
1160  FOR TO = 1 TO FM
1165  IF TO = 1 THEN 1200
1170 T1 = T1 + T2
1175 T3 = T3 +  INT (T1 / 60)
1180 T1 = T1 - 60 *  INT (T1 / 60)
1185  IF T3 <  = 59 THEN 1200
1190 T4 = T4 + 1
1195 T3 = T3 - 60
1200 T$(TO) =  RIGHT$ ( STR$ (100 + T4),2)
1205 T$(TO) = T$(TO) +  RIGHT$ ( STR$ (100 + T3),2)
1210 T$(TO) = T$(TO) +  RIGHT$ ( STR$ (100 + T1),2)
1215  REM  T$(I) IS 6 CHARACTERS WITH HRMNSC
1220  NEXT
1300  REM  NOW MAKE UP THE ALTITUDE STRINGS
1305 SS =  VAL ( MID$ (B$,40,2))
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.10 cont'd.

```
1310  FOR I = 1 TO 20
1315 HT$(I) =  STR$ ((SS - 1 + I) * 1.5)
1320  NEXT I
1325  REM  ALL INFORMATION HAS BEEN TAKEN FROM B$
1400  REM  GET THE DC VALUES
1405  REM  GET THE REALS
1407 CT = 1
1410  FOR I = 1 TO 7 STEP 2
1415  FOR J = 1 TO 33 STEP 8
1420 DR$(CT) =  STR$ ( VAL ( MID$ (A$(I),J,8)))
1425 DI$(CT) =  STR$ ( VAL ( MID$ (A$(I + 1),J,8)))
1430 CT = CT + 1
1435  NEXT
1440  NEXT
1445  REM  DC NOW IN ARRAYS, DONE WITH A$(I) STUFF
1450  REM  DONE WITH ONE TIME ONLY STUFF
1500  REM  SET FLAG FOR ERROR CONTROL
1505 ST = 0
1510  GOTO 5000
1520  REM  GOT VOLUME OK, GET READY FOR FFTS
1521  REM  MODIFY ERROR CONTROL FLAG
1522 ST = 1
1525  REM  FORCE GARBAGE COLLECTION
1530 B$ = S$ + S$
1535  FOR I = 1 TO 8:A$(I) = S$ + S$: NEXT
1540 I =  FRE (0)
1545  REM  THE FOLLOWING LOOPS ARE NOT FOR-NEXT
1550  REM  SO THAT REENTRY AFTER ERROR CAN OCCUR
1567 N6 = N3 / 16
1570  REM  SIGNAL THAT APPLE IS READY
1575  FOR I = 1 TO 10: PRINT  CHR$ (7): NEXT
1580  HOME : PRINT "WAITING FOR FFT DATA"
2000  REM  READ FFT FROM PDP-15
2001  REM  GET HGTNO,FFTNO,FUDGE FACTOR, THEN ARRAY
2005  CALL 832
2007 B$ = M$ + S$
2010  FOR I = 1 TO N6
2015  CALL 832
2020 A$(I) = M$ + S$
2025  NEXT
2030  REM  GET HGTNO,FFTNO,FUDGE FACTOR
2035 HN =  VAL ( MID$ (B$,7,2))
2040 NF =  VAL ( MID$ (B$,15,2))
2045 FF$ =  STR$ ( VAL ( MID$ (B$,17,12)))
2100  REM  BEGIN WRITE TO APPLE DISK
2105  REM  MAKE FILE NAME
2110 F$ = "FFTA/" + T$(NF) + "/" + HT$(HN)
2112  HOME : PRINT "WRITING FILE ";F$
2115  REM  REMOVE OLD ONE
2117  ONERR  GOTO 4210
2120  PRINT D$;"OPEN ";F$
2125  PRINT D$;"DELETE ";F$
2127  ONERR  GOTO 4200
2130  PRINT D$;"OPEN ";F$
2135  PRINT D$;"WRITE ";F$
2140  REM  SEND DATE WITH DOT, TYPE OF DATA, TIME, ALTITUDE
2145  PRINT H$
2150  PRINT "FFTA"
2155  PRINT T$(NF)
2160  PRINT HT$(HN)
2165  REM  NOW SEND NO. OF POINTS, FUDGE FACTOR, DC VALUES
2170  PRINT  STR$ (N3)
2175  PRINT FF$
2180  PRINT DR$(HN)
2185  PRINT DI$(HN)
2190  REM  SEND ARRAY,STRIP LEADING SPACES
2195  FOR I = 1 TO N6
2200  FOR J = 0 TO 45 STEP 3
2205 CT = 1
```

Table A.10 cont'd.

```
2210  IF  MID$ (A$(I),J + CT,1) = S$ THEN CT = CT + 1: GOTO 2210
2215  PRINT  MID$ (A$(I),J + CT,4 - CT)
2220  NEXT
2225  NEXT
2230  PRINT D$;"CLOSE ";F$
2232  IF NF = FM AND HN = 20 THEN 2300
2235  REM  DONE WITH DISK, CLEAN HOUSE
2240  FOR I = 1 TO N6
2245 A$(I) = S$ + S$
2250  NEXT I
2255 B$ = S$ + S$
2260 I =  FRE (0)
2265  REM  READY FOR NEXT ONE, LOOP BACK
2270  GOTO 1570
2300  REM  DONE
2305  PRINT "& AND RETURN TO RESTART"
2307  POKE 216,0
2310  END
4200  REM  NO ROOM ON DISK
4205  PRINT D$;"DELETE ";F$
4210  PRINT : PRINT "THIS DISK DOES NOT HAVE ENOUGH ROOM"
4215  PRINT "FOR THE ENTIRE FILE-IT HAS BEEN REMOVED."
4220  PRINT : PRINT "MAKE ANOTHER DISK WITH CORRECT HEADER."
4225  GOTO 4600
4500  REM  ERROR WHILE CHECKING DISK
4510  PRINT : PRINT "DISK AT GIVEN S,D,V NO GOOD"
4520  PRINT "ONE OF SEVERAL ERRORS HAS OCCURED:"
4530  PRINT : PRINT "  1. S,D,V GIVEN DO NOT POINT TO A DISK"
4535  PRINT "  2. FLOPPY NOT IN DRIVE, OR BLANK"
4540  PRINT "  3. ACCESS DENIED TO THIS VOLUME"
4542  PRINT "  4. DISK WRITE PROTECTED"
4544  PRINT "  5. DISK FULL"
4550  PRINT
4600  REM  DISK READY ROUTINE
4610  PRINT : PRINT "EITHER INSERT CORRECT FLOPPY INTO DRIVE"
4620  PRINT "OR PREPARE A CORVUS VOLUME."
4630  PRINT : PRINT "IF CORRECT VOLUME NOW AVAILABLE"
4640  PRINT "SOMEWHERE ON CORVUS OR IN FLOPPY DRIVE"
4650  PRINT "TYPE C FOR CONTINUE.  OTHERWISE"
4660  PRINT "TYPE Q FOR QUIT."
4670  INPUT "Q)UIT OR C)ONTINUE ? ";B$
4680  IF  LEFT$ (B$,1) = "Q" GOTO 2300
4690  IF  LEFT$ (B$,1) = "C" GOTO 5000
4695  GOTO 4670
5000  REM  CHECK DOS POINTERS
5010  PRINT : PRINT "DOS POINTERS SET TO S";SL;",D";DR;",V";VL
5020  PRINT : INPUT "CHANGE THESE? Y=YES, N=NO ";B$
5030  IF  LEFT$ (B$,1) = "N" GOTO 6000
5035  IF  LEFT$ (B$,1) = "Y" GOTO 5040
5037  GOTO 5020
5040  PRINT : INPUT "SLOT,DRIVE,VOLUME? ";SL,DR,VL
5050  IF DR <  = 2 AND DR >  = 1 AND SL <  = 6 AND SL >  = 4 AND VL >  =
     0 GOTO 6000
5060  PRINT "S=4,5,6; D=1,2; V>=0"
5070  GOTO 5040
6000  REM  DISK CHECK ROUITNE
6010  ONERR  GOTO 4500
6020  PRINT D$;"OPEN DUMMY,S"; STR$ (SL);",D"; STR$ (DR);",V"; STR$ (VL)
6030  PRINT D$;"DELETE DUMMY"
6040  REM  CHECK HEADER
6050  ONERR  GOTO 7000
6060  PRINT D$;"VERIFY ";H$
6070  IF ST = 0 GOTO 1520
6080  IF ST = 1 GOTO 2115
6090  PRINT "ERROR IN FLAG HANDLING"
6100  STOP
7000  REM  HEADER MISMATCH
7010  PRINT "HEADER MISMATCH"
7020  GOTO 4600
```

The transfer of FFTs to the Apple II is fundamentally the same as the transfer for the minute-by-minute data which is described by Roth [1982]. Strings of 48 characters are sent to the PDP-15 terminal which the Apple II is monitoring. The Apple II reads the string into high memory with a machine-language routine and then quickly copies the string to another location and returns to wait for the next string from the PDP-15. After a group of strings, i.e., one FFT, has been transferred, the Apple II parses the strings which it has stored and writes the FFT values to disk. The FFT strings are read in lines 2000 to 2025, and the write to disk occurs in lines 2100 to 2230. Note that the program writes over old versions of a file with the same name and checks various conditions such as disk full, wrong data file on disk, etc., in such a way that the program flow can be reentered. This prevents the use of FOR-NEXT loops. The program will not lose the transferred data if a floppy disk becomes full -- it prompts for a new disk and when it has verified the correct date on this disk the FFT is stored. The program will exit after it stores the last FFT from the PDP-15. The number of FFTs at each height is determined in line 1150 from information in the header, but the number of heights is fixed at 20. This could be modified to allow the processing of fewer FFTs.

The file structure for the FFT files is shown in Table A.11. The data are stored with a fixed format header followed by an array whose length varies with the number of points in the FFT. The number of points is therefore included in the header. Note that the scaling factor in the header multiplied by the integer in the array gives the actual value for that component of the FFT. The elements are stored with the zero-frequency value first and the highest frequency value last. The last half of the array should be wrapped around to display positive and negative frequencies.

Table A.11 Apple II disk file format for FFT files.

Sequential text file, Apple DOS 3.3

File name e.g. FFTA/083230/67.5

H$: date with a dot e.g., "November 13. 1982"

"FFTA": indicates the file type "FFTA" for 32-, 64- and 128-second FFTs, "FFT8" for 8 second FFTs

T$: time of the spectrum e.g., "092430"

HT: altitude of the spectrum in km e.g., 85.5

N3: number of points in the spectrum

FF: scaling factor for the data in the file

DR: DC offset for the real channel at this altitude

DI: DC offset for the imaginary channel at this altitude

A(1): value of the power spectrum at zero frequency -- an integer from 0 to 999

A(N3): value at the highest frequency

The AP15DRIVER program shown in Table A.12 is used for all transfers between the PDP-15 and the Apple II. The section beginning in location $300 is for the control of the Apple II serial card. This code, which runs in place of the code in ROM on the card, is faster because it is specific to the baud rate and direction of transfer. The routine at $340 is called from BASIC to read a single line of 48 characters preceded by several asterisks. It uses the routine at $300 to read each character. The asterisks provide time for the Apple II to resynchronize with the PDP-15 character stream and signal the Apple II that the characters which follow are data being transferred. The loop between $343 and $34B reads characters until an asterisk is found. The loop between then reads characters until a non-asterisk is found. That character and the 47 which follow are stored in the memory set aside by the M$ declaration in the FFTTOAPDISKSRC program. The number of characters to be read is determined by the value loaded into the Y register in $357 and must agree with the length of the string area set aside. It should be clear that it is the programmers job to insure that the interpretation of the characters in the string is the same in the Apple II and the PDP-15.

A.5 Processing and Transfer of 8-Second Spectra

The programs used for the calculation and transfer of 8-second power spectra are similar to those shown above for the 32-, 64-, and 128-second spectra. Because of the large number of 8-second spectra which can be calculated from a data tape of 400 seconds and 20 altitudes, however, the 8-second programs allow direct control of which FFTs are calculated. Furthermore, the 8-second interval is less than the 10-second length of a record so that no overlap of the data is required and each record is read only once from the data file. Finally, the PDP-15 must signal the Apple II

Table A.12 AP15DRIVER.


```
0300-   78          SEI
0301-   A9 09       LDA   #$09
0303-   85 35       STA   $35
0305-   38          SEC
0306-   B0 05       BCS   $030D
0308-   C6 35       DEC   $35
030A-   F0 1C       BEQ   $0328
030C-   18          CLC
030D-   AD 90 C0    LDA   $C090
0310-   90 0B       BCC   $031D
0312-   10 02       BPL   $0316
0314-   30 F7       BMI   $030D
0316-   A0 19       LDY   #$19
0318-   88          DEY
0319-   D0 FD       BNE   $0318
031B-   F0 EB       BEQ   $0308
031D-   2A          ROL
031E-   6E 79 06    ROR   $0679
0321-   A0 0F       LDY   #$0F
0323-   88          DEY
0324-   D0 FD       BNE   $0323
0326-   F0 E0       BEQ   $0308
0328-   AD 90 C0    LDA   $C090
032B-   10 FB       BPL   $0328
032D-   60          RTS
032E-   00          BRK
032F-   00          BRK
0330-   00          BRK
0331-   00          BRK
0332-   00          BRK
0333-   00          BRK
0334-   00          BRK
0335-   00          BRK
0336-   00          BRK
0337-   00          BRK
0338-   00          BRK
0339-   00          BRK
033A-   00          BRK
033B-   00          BRK
033C-   00          BRK
033D-   00          BRK
033E-   00          BRK
033F-   00          BRK
0340-   20 4A FF    JSR   $FF4A
0343-   20 00 03    JSR   $0300
0346-   AD 79 06    LDA   $0679
0349-   C9 2A       CMP   #$2A
034B-   D0 F6       BNE   $0343
034D-   20 00 03    JSR   $0300
0350-   AD 79 06    LDA   $0679
0353-   C9 2A       CMP   #$2A
0355-   F0 F6       BEQ   $034D
0357-   A0 30       LDY   #$30
0359-   A2 00       LDX   #$00
035B-   AD 79 06    LDA   $0679
035E-   9D CF 95    STA   $95CF,X
0361-   E8          INX
0362-   88          DEY
0363-   F0 0A       BEQ   $036F
0365-   98          TYA
0366-   48          PHA
0367-   20 00 03    JSR   $0300
036A-   68          PLA
036B-   A8          TAY
036C-   4C 5B 03    JMP   $035B
036F-   20 3F FF    JSR   $FF3F
0372-   60          RTS
```

to quit since the number of FFTs can vary. These changes are discussed below.

The PDP-15 program FFT8MN, shown in Table A.13, calculates the 8-second spectra and transfers the data to the Apple II. This program is virtually identical to FFTAMN through the calculation of the dc averages. Slight changes are necessary because this program deals with a fixed length spectrum. Beginning with line 30 the operator chooses which spectrum to calculate. First, the operator inputs a record number from 1 to 40. An input of 0 for the record number ends the program. Note that the program will backup, i.e., read in a record previous to the one just used. However, this requires closing the open data file, reading in all the records preceding the desired one, and then ignoring them. It is therefore faster to calculate the spectra in chronological order. When the correct record is in memory the operator is prompted for the height number of interest. An input of 0 at this point returns the program to the record number input sequence and therefore should be specified after all heights of interest for that record have been processed.

The RE and AIM arrays used in the FFT calculation are filled directly from the array read from the tape or disk instead of a temporary array as in FFTAMN. The routine FTFILL is therefore not necessary for FFT8MN and should not appear in the loader command string. The programs FTCALC and FTREAD are still necessary however, and the assignment statement is unchanged. As in FFTAMN the data are transferred in an integer array, but the header information must be more complete for the FFT8MN transfer because the spectra may be calculated in any order. The PDP-15 waits for a short period after sending the header so that the Apple II will have time to determine the

Table A.13 FFT8MN.


```
C  THIS PROGRAM CALCULATES 8 SECOND FFT FROM CINT DATA
C  COPY DATA FROM DECTAPE TO DISK USING PIP-IMAGE BINARY
C  ASSIGN TT TO SLOT 2
C  ASSIGN DISK WITH DATA TO SLOT 6
C  DATA COLLECTED WITH SCAT AND CM PROGRAMS
C  USES EXTERNAL MACRO ROUTINE FTREAD
C  USES EXTERNAL FORTRAN ROUTINES FTCALC
C  KEN GIBBS  6/18/82
         REAL RE(1024),AIM(1024),COR(512),COI(512),FILEN(2)
         INTEGER IN(3203),OUT(64),DR(20),DI(20)
         LOGICAL OK
         DIMENSION CO(20),CI(20)
         COMMON /A/IN/F/N,M,NV2,NM1,RE,AIM,COR,COI
C  WAKE UP THE DECWRITER
         WRITE(2,66)
66       FORMAT(1X,2HH )
C  GET FILE NAME AND CHECK FOR MATCH
70       WRITE(2,67)
67       FORMAT(1X,30HINPUT FILE NAME E.G. 14MAY 80A)
         READ(2,68) FILEN(1),FILEN(2)
68       FORMAT(A5,A4)
         CALL FSTAT(6,FILEN,OK)
         IF (OK) GO TO 84
         WRITE(2,69)
69       FORMAT (1X,24HMISMATCH...PLEASE RETYPE)
         GO TO 70
74       FORMAT(I2)
C  NOW CALCULATE A START TIME BASED ON WHICH TAPE IN A SERIES
C  THIS DATA IS AND THE START TIME OF FIRST TAPE
84       WRITE(2,85)
85       FORMAT(1X,43HIS THIS THE 1ST, 2ND, OR 3RD TAPE..2 DIGITS)
         READ(2,74) ITAPE
         IF (ITAPE.LT.1.OR.ITAPE.GT.4) GO TO 84
86       WRITE(2,87)
87       FORMAT(1X,42HINPUT START TIME OF FIRST TAPE IN THIS SET)
88       WRITE(2,89)
89       FORMAT(1X,23HHOURS-FROM 00 TO 23-CST)
         READ(2,74) IHOUR
         IF (IHOUR.LT.0.OR.IHOUR.GT.23) GO TO 89
90       WRITE(2,91)
91       FORMAT(1X,21HMINUTES-FROM 00 TO 59)
         READ(2,74)MINUT
         IF (MINUT.LT.0.OR.MINUT.GT.59) GO TO 91
C  CALCULATION OF TIME BASED ON THE ABOVE
         ISCND=400*(ITAPE-1)
         MINUT=MINUT+ISCND/60
         IF (MINUT.LT.59) GO TO 92
         MINUT=MINUT-60
         IHOUR=IHOUR+1
92       ISCND=ISCND-60*(ISCND/60)
C  NOW THE START HEIGHT INFO
94       WRITE (2,93)
93       FORMAT(1X,28HINPUT START SETTING-2 DIGITS)
         READ(2,74) ISS
         IF (ISS.LT.30.OR.ISS.GT.60) GO TO 94
C  SET UP PARAMETERS FOR FFT
         ISEC=10
         M=6
         N=2**M
         NV2=N/2
         NM1=N-1
         NV16=N/16
         ZERO=0.0
         IRCOL=41
         IWAIT=10000
C  FILL COEFFECIENT ARRAYS
         PI=3.14159265
         F1=PI/FLOAT(NV2)
         ANGLE=0.0
```

Table A.13 cont'd.

```
        DO 400 I=1,NV2
        COR(I)=COS(ANGLE)
        COI(I)=-SIN(ANGLE)
400     ANGLE=ANGLE+F1
C  SET UP READ ROUTINE WITH ADDRESS OF IN ARRAY
        CALL SET(IN)
C  FIND DC AVERAGES
        CALL SEEK(6,FILEN)
        DO 105 I=1,20
        CO(I)=0.0
105     CI(I)=0.0
        DO 79 I=1,40
        CALL GRAB
        IF (IN(3202).EQ.0) IROFF=20
        IF (IN(3202).NE.0) IROFF=0
        IIMOF=20-IROFF
        DO 79 JOFF=1,80
        DO 80 IHGT=1,20
        IROFF=IROFF+1
        IIMOF=IIMOF+1
        CO(IHGT)=CO(IHGT)+FLOAT(IN(IROFF))
80      CI(IHGT)=CI(IHGT)+FLOAT(IN(IIMOF))
        IIMOF=IIMOF+20
79      IROFF=IROFF+20
        CALL CLOSE(6)
        DO 106 IHGT=1,20
        DR(IHGT)=IFIX(CO(IHGT)/3200.)
106     DI(IHGT)=IFIX(CI(IHGT)/3200.)
C  DONE WITH DC AVERAGE
30      WRITE(2,36)
36      FORMAT(1X,35HINPUT RECORD NO. 1 TO 40, 0 TO QUIT)
        READ(2,74)IRCNW
        IF(IRCNW.LT.0.OR.IRCNW.GT.40) GOTO 30
        IF(IRCNW.EQ.0) GOTO 500
31      IF (IRCNW.EQ.IRCOL) GOTO 33
        IF (IRCNW.GT.IRCOL) GOTO 32
        CALL SEEK(6,FILEN)
        IRCOL=0
32      CALL GRAB
        IRCOL=IRCOL+1
        GOTO 31
C  NOW GET HEIGHT
33      WRITE(2,35)
35      FORMAT(1X,39HINPUT HGT NO. 1 TO 20, 0 FOR NEW RECORD)
        READ(2,74) IHGT
        IF (IHGT.LT.0.OR.IHGT.GT.20) GOTO 33
        IF (IHGT.EQ.0) GOTO 30
        HGT=FLOAT(ISS+IHGT-1)*1.5
C  START CALCULATION
        IF (IN(3202).EQ.0) IROFF=20
        IF (IN(3202).NE.0) IROFF=0
        IIMOF=20-IROFF+IHGT
        IROFF=IROFF+IHGT
C  ONLY INTERESTED IN FIRST 8 SECONDS N=64
        DO 34 I=1,N
        RE(I)=FLOAT(IN(IROFF)-DR(IHGT))
        AIM(I)=FLOAT(IN(IIMOF)-DI(IHGT))
        IIMOF=IIMOF+40
34      IROFF=IROFF+40
C  TRANSFORM
        CALL FFT
C  FIND POWER SPECTRUM,NORMALIZE,FIX
        HIGH=RE(1)
        DO 600 I=1,N
        RE(I)=RE(I)*RE(I)+AIM(I)*AIM(I)
600     IF (HIGH.LT.RE(I)) HIGH=RE(I)
        FUDGE=999.0/HIGH
        DO 601 I=1,N
601     OUT(I)=IFIX(RE(I)*FUDGE)
```

Table A.13 cont'd.

```
C  CALCUTE TIME FOR THIS RECORD
       IHR2=IHOUR
       ISCAD=(IRCNW-1)*10
       ISEC2=ISCND+ISCAD
       MINAD=ISEC2/60
       ISEC2=MOD(ISEC2,60)
       MINT2=MINUT+MINAD
       IF (MINT2.LE.59) GOTO 700
       MINT2=MINT2-60
       IHR2=IHR2+1
C  DONE WITH TIME
700    WRITE(2,701) IHR2,MINT2,ISEC2,HGT,N,FUDGE,
    1  FILEN(1),FILEN(2),DR(IHGT),DI(IHGT)
701    FORMAT(1X,8H********,3I2,F5.1,I3,E12.5,2A5,2I6,1H*)
C  WAIT FOR APPLE TO DIGEST THIS
       DO 703 I=1,IWAIT
703    CONTINUE
       WRITE(2,702) (OUT(J),J=1,16)
       WRITE(2,702) (OUT(J),J=17,32)
       WRITE(2,702) (OUT(J),J=33,48)
       WRITE(2,702) (OUT(J),J=49,64)
702    FORMAT(1X,8H********,16I3,1H*)
       GOTO 33
500    CALL CLOSE(6)
C  DONE WITH DATA, INFORM USER, QUIT
       WRITE(2,550)
550    FORMAT(1X,8H********,8HALL DONE,5X,1H0,34X,1H*)
       STOP
       END
```

number of points in the FFT. A header sent with a 0 for the number of points signals the Apple II to quit.

The program FFT8TOAPDISK2SRC, shown in Table A.14, stores the 8-second power spectra on Apple disk with the file structure shown above in Table A.11. The changes in this program parallel the changes in the PDP-15 program discussed above. The variable PS is used as a flag to indicate the number of passes through the main loop between lines 1000 and 2270. The date string is formed from the transferred header only on the first pass. The date file on the Apple disk is then checked for agreement. The disk check routines are not executed again until a disk is filled and a new disk is started. The main loop is then reentered and processing resumed. FFT8TOAPDISK2SRC should be used in its compiled version, FFT8TOAPDISK2.OBJ, in exactly the same manner as given above for FFTTOAPDISK2.OBJ.

A.6 Plotting of 8-, 32-, 64-, and 128-Second Spectra

The plotting of FFT data from Apple disks requires the FFTTOHP2SRC program for the Apple II, shown in Table A.15, and the HP-9830 FFTA plot program shown in Table A.16. Both of these programs work for the FFTA files and the FFT8 files. The compiled version of the Apple II program, FFTTOHP2.OBJ, should be used to increase the speed of the transfer to the HP-9830. The compiler runtime library should be loaded first with the sequence BLOAD RUNTIME, BRUN FFTTOHP2.OBJ.

The Apple II is used as an inteliigent disk drive for the HP-9830 with data transferred as ASCII characters via an 8-bit parallel link described by Roth [1982]. Lines 10 to 80 are used to read a file from Apple disk. It is important to verify that a file exists before attempting to open it in order to avoid generating files on disks which are not write protected. The entire file is read in and then the file is closed. Lines 100 to 125 set up

Table A.14 FFT8TOAPDISK2SRC.

ORIGINAL PAGE IS
OF POOR QUALITY

```
1 REM TASC COMPILER SOURCE ONLY
2 REM WILL NOT RUN WITH AP15DRIVER
3 REM UNDER THE INTERPRETER
4 REM KEN GIBBS 7/3/82
5 REM ! INTEGER I,TO,CT,J,N6
6 REM ! INTEGER SL,DR,VL,PS
10 REM PROGRAM FOR TRANSFER OF 8 SECOND FFTS OF COHERENTLY INTEGRATED D
   ATA
11 REM USES SAME FILE STRUCTURE AS FOR
12 REM 64 SEC. FFTS, BUT HAS LESS DATA.
100 M$ = "1234567890123456789012345678901234567890l234567"
101 M$ = M$ + "8"
102 S$ = " ":D$ = CHR$ (4)
103 SL = 6:DR = 1:VL = 0
105 DIM E$(12),A$(4)
110 REM SET UP MONTH TABLE
111 FOR I = 1 TO 12: READ E$(I): NEXT
112 DATA "JANUARY","FEBRUARY","MARCH","APRIL","MAY"
113 DATA "JUNE","JULY","AUGUST","SEPTEMBER"
114 DATA "OCTOBER","NOVEMBER","DECEMBER"
120 PS = 0
1000 PRINT : PRINT "WAITING FOR DATA"
1001 REM GET TIME,HEIGHT,NO. OF POINTS,
1002 REM FUDGE FACTOR, FILE NAME
1005 CALL 832
1007 B$ = M$ + S$
1008 N3 = VAL ( MID$ (B$,12,3))
1009 IF N3 = 0 GOTO 2300
1010 N6 = N3 / 16: REM NO. OF DATA STRINGS
1011 FOR I = 1 TO N6
1015 CALL 832
1020 A$(I) = M$ + S$
1025 NEXT I
1027 PRINT : PRINT "GOT FFT"
1028 PS = PS + 1: REM INCREMENT NO. OF PASSES
1030 IF PS > 1 GOTO 1100: REM DATE FIRST TIME ONLY
1031 REM SCAN THIRD CHARACTER OF MONTH
1040 FOR I = 1 TO 12
1045 IF MID$ (B$,31,1) = MID$ (E$(I),3,1) THEN MN$ = E$(I): GOTO 1060
1050 NEXT
1055 PRINT "FILENAME ERROR": STOP
1060 REM CHECK SECOND LETTER FOR SPECIAL CASES
1065 REM LOOP ABOVE PICKS FIRST MATCH, SWITCH TO SECOND IF NECESSARY
1070 IF MID$ (B$,31,1) = "N" AND MID$ (B$,30,1) = "U" THEN MN$ = E$(6)
     : GOTO 1080
1075 IF MID$ (B$,31,1) = "R" AND MID$ (B$,30,1) = "P" THEN MN$ = E$(4)

1080 H$ = MN$ + S$ + STR$ ( VAL ( MID$ (B$,27,2))) + "." + S$ + "19" + MID$
     (B$,33,2)
1085 REM H$=DATE WITH DOT AS STRING=HEADER FILE NAME
1100 REM NOW FIX THE TIME
1105 REM GET HOURS,MINUTES,SECONDS
1110 T4 = VAL ( LEFT$ (B$,2))
1115 T3 = VAL ( MID$ (B$,3,2))
1120 T1 = VAL ( MID$ (B$,5,2))
1200 T$ = RIGHT$ ( STR$ (100 + T4),2)
1205 T$ = T$ + RIGHT$ ( STR$ (100 + T3),2)
1210 T$ = T$ + RIGHT$ ( STR$ (100 + T1),2)
1215 REM T$ IS 6 CHARACTERS WITH HRMNSC
1300 REM NOW THE ALTITUDE
1305 HT$ = STR$ ( VAL ( MID$ (B$,7,5)))
1310 REM NOW THE FUDGE FACTOR
1315 FF$ = STR$ ( VAL ( MID$ (B$,15,12)))
1400 REM GET THE DC VALUES
1405 DR$ = STR$ ( VAL ( MID$ (B$,37,6)))
1410 DI$ = STR$ ( VAL ( MID$ (B$,43,6)))
1500 REM CHECK DISK FIRST TIME
1505 IF PS > 1 GOTO 1520
1510 GOTO 5000: REM CHECK HEADER,NOT WRITE PROTECTED
```

Table A.14 cont'd.


```
1520  REM  DISK OK
2100  REM  BEGIN WRITE TO APPLE DISK
2105  REM  MAKE FILE NAME
2110 F$ = "FFT8/" + T$ + "/" + HT$
2112  HOME : PRINT "WRITING FILE ";F$
2115  REM  REMOVE OLD ONE
2117  ONERR  GOTO 4210
2120  PRINT D$;"OPEN ";F$
2125  PRINT D$;"DELETE ";F$
2127  ONERR  GOTO 4200
2130  PRINT D$;"OPEN ";F$
2135  PRINT D$;"WRITE ";F$
2140  REM  SEND DATE WITH DOT, TYPE OF DATA, TIME, ALTITUDE
2145  PRINT H$
2150  PRINT "FFT8"
2155  PRINT T$
2160  PRINT HT$
2165  REM  NOW SEND NO. OF POINTS, FUDGE FACTOR, DC VALUES
2170  PRINT N3
2175  PRINT FF$
2180  PRINT DR$
2185  PRINT DI$
2190  REM  SEND ARRAY,STRIP LEADING SPACES
2195  FOR I = 1 TO N6
2200  FOR J = 0 TO 45 STEP 3
2205 CT = 1
2210  IF  MID$ (A$(I),J + CT,1) = S$ THEN CT = CT + 1: GOTO 2210
2215  PRINT  MID$ (A$(I),J + CT,4 - CT)
2220  NEXT
2225  NEXT
2230  PRINT D$;"CLOSE ";F$
2235  REM  DONE WITH DISK, CLEAN HOUSE
2240  FOR I = 1 TO N6
2245 A$(I) = S$ + S$
2250  NEXT I
2255 B$ = S$ + S$
2260 I =  FRE (0)
2265  REM  READY FOR NEXT ONE, LOOP BACK
2270  GOTO 1000
2300  REM  DONE
2305  PRINT "& AND RETURN TO RESTART"
2307  POKE 216,0
2310  END
4200  REM  NO ROOM ON DISK
4205  PRINT D$;"DELETE ";F$
4210  PRINT : PRINT "THIS DISK DOES NOT HAVE ENOUGH ROOM"
4215  PRINT "FOR THE ENTIRE FILE-IT HAS BEEN REMOVED."
4220  PRINT : PRINT "MAKE ANOTHER DISK WITH CORRECT HEADER."
4225  GOTO 4600
4500  REM  ERROR WHILE CHECKING DISK
4510  PRINT : PRINT "DISK AT GIVEN S,D,V NO GOOD"
4520  PRINT "ONE OF SEVERAL ERRORS HAS OCCURED:"
4530  PRINT : PRINT "  1. S,D,V GIVEN DO NOT POINT TO A DISK"
4535  PRINT "  2. FLOPPY NOT IN DRIVE, OR BLANK"
4540  PRINT "  3. ACCESS DENIED TO THIS VOLUME"
4542  PRINT "  4. DISK WRITE PROTECTED"
4544  PRINT "  5. DISK FULL"
4550  PRINT
4600  REM  DISK READY ROUTINE
4610  PRINT : PRINT "EITHER INSERT CORRECT FLOPPY INTO DRIVE"
4620  PRINT "OR PREPARE A CORVUS VOLUME."
4630  PRINT : PRINT "IF CORRECT VOLUME NOW AVAILABLE"
4640  PRINT "SOMEWHERE ON CORVUS OR IN FLOPPY DRIVE"
4650  PRINT "TYPE C FOR CONTINUE.  OTHERWISE"
4660  PRINT "TYPE Q FOR QUIT."
4670  INPUT "Q)UIT OR C)ONTINUE ? ";B$
4680  IF  LEFT$ (B$,1) = "Q" GOTO 2300
4690  IF  LEFT$ (B$,1) = "C" GOTO 5000
4695  GOTO 4670
```

Table A.14 cont'd.

```
5000  REM  CHECK DOS POINTERS
5010  PRINT : PRINT "DOS POINTERS SET TO S";SL;",D";DR;",V";VL
5020  PRINT : INPUT "CHANGE THESE? Y=YES, N=NO ";B$
5030  IF  LEFT$ (B$,1) = "N" GOTO 6000
5035  IF  LEFT$ (B$,1) = "Y" GOTO 5040
5037  GOTO 5020
5040  PRINT : INPUT "SLOT,DRIVE,VOLUME? ";SL,DR,VL
5050  IF DR <  = 2 AND DR >  = 1 AND SL <  = 6 AND SL >  = 4 AND VL >  =
     0 GOTO 6000
5060  PRINT "S=4,5,6; D=1,2; V>=0"
5070  GOTO 5040
6000  REM  DISK CHECK ROUITNE
6010  ONERR  GOTO 4500
6020  PRINT D$;"OPEN DUMMY,S"; STR$ (SL ·",D"; STR$ (DR);",V"; STR$ (VL)
6030  PRINT D$;"DELETE DUMMY"
6040  REM  CHECK HEADER
6050  ONERR  GOTO 7000
6060  PRINT D$;"VERIFY ";H$
6080  GOTO 1520
7000  REM  HEADER MISMATCH
7010  PRINT "HEADER MISMATCH"
7020  GOTO 4600
```

ORIGINAL PAGE IS OF POOR QUALITY

Table A.15 FFTTOHP2SRC.

```
 LOAD FFTTOHP2SRC
]LIST

1  REM  THIS PROGRAM READS T' . FFTA FILES
2  REM  AND TRANSFERS THEM TC HP FOR PLOT
5  REM  KEN GIBBS 8/4/82 TASC SOURCE
6  REM  REQUIRES HP CARD IN SLOT 3
9  REM  ! INTEGER I,J,A1,A2,N3
10 D$ =  CHR$ (4): DIM A$(1024),B$(8)
12  PRINT : PRINT
13  PRINT "C)ATALOG F)ILE INPUT Q)UIT"
14  INPUT "INPUT C,F, OR Q ";Q$
15  IF  LEFT$ (Q$,1) = "Q" GOTO 260
20  IF  LEFT$ (Q$,1) = "F" GOTO 35
25  IF  LEFT$ (Q$,1) = "C" GOTO 290
30  GOTO 13
35  INPUT "FILE NAME ? ";F$
40  ONERR  GOTO 280
41  PRINT D$;"VERIFY ";F$: POKE 216,0
45  PRINT D$;"OPEN ";F$: PRINT D$;"READ ";F$
46  REM  GET HEADER STUFF
50  FOR I = 1 TO 8: INPUT B$(I): NEXT
51  REM  GET DATA ARRAY
52  REM  FIRST, HOW MANY NUMBERS
53 N3 =  VAL (B$(5))
55  FOR I = 1 TO N3
60  INPUT A$(I)
65  NEXT
80  PRINT D$;"CLOSE ";F$
100  REM  SET UP VIA
105  POKE  - 15486,255: REM  DDRB=$FF OUTPUT
110  POKE  - 15476,136: REM  PCR=$88 HANDSHAKE
115  POKE  - 15475,255: REM  CLEAR IFR
120  POKE  - 15474,127: REM  CLEAR IER
125  REM  -15488=B DATA REGISTER
130  PRINT : PRINT "START HP PROGRAM NOW"
132  REM  SEND HEADER
135  FOR I = 1 TO 8
140 A1 =  LEN (B$(I))
145  FOR A2 = 1 TO A1
150  IF  PEEK ( - 15475) <  > 18 THEN 150
155  POKE  - 15488, ASC ( MID$ (B$(I),A2,1))
160  NEXT
165  IF  PEEK ( - 15475) <  > 18 THEN 165
170  POKE  - 15488,10
175  NEXT
180  REM  SEND ARRAY
190  FOR J = 1 TO N3
200 A1 =  LEN (A$(J))
205  FOR A2 = 1 TO A1
210  IF  PEEK ( - 15475) <  > 18 THEN 210
215  POKE  - 15488, ASC ( MID$ (A$(J),A2,1))
220  NEXT
225  IF  PEEK ( - 15475) <  > 18 THEN 225
230  POKE  - 15488,10
235  NEXT
245  REM  DONE-RESTART
250  RUN
255  STOP
260  PRINT "& AND RETURN TO RESTART"
265  END
270  PRINT "ERROR TRYING TO CATALOG"
275  GOTO 13
280  PRINT "NO SUCH FILE ON DISK"
285  GOTO 13
290  ONERR  GOTO 270
295  PRINT D$;"CATALOG"
300  GOTO 13
```

Table A.16 HP-9830 FFTA plot program.


```
10 REM PROGRAM TO PLOT FFTA STUFF
11 REM ALSO WORKS FOR FFTB
20 REM KEN GIBES 6/13/82
30 REM PROGRAM SHOULD BE RUN WHEN INDICATED BY APPLE
40 DIM D[4,256],P[40],L$[60],U$[20],F$[25]
50 DIM G$[25],T$[6],H$[16]
60 DIM F[10]
70 FOR I=1 TO 5
72 READ F[I]
74 NEXT I
76 DATA -1.6,-2,-2.3,-2.3,-2
100 REM GET HEADER STUFF
105 REM DATE WITH DOT
110 ENTER (2,*)G$
112 REM GET FFTA STRING
113 ENTER (2,*)L$
115 REM TIME AS STRING
120 ENTER (2,*)T$
125 REM HEIGHT AS STRING
130 ENTER (2,*)H$
135 REM NO. OF POINTS
140 ENTER (2,*)N3
145 REM FUDGE FACTOR
150 ENTER (2,*)F1
155 REM DC OFFSET, BOTH CHANNELS
160 ENTER (2,*)D0
165 ENTER (2,*)D1
200 REM GET DATA
201 REM IF N3>256 FOLD ARRAY
202 IF N3>256 THEN 210
203 J1=N3
204 I1=1
205 GOTO 215
210 I1=N3/256
211 J1=256
215 FOR I=1 TO I1
220 FOR J=1 TO J1
225 ENTER (2,*)D[I,J]
230 NEXT J
235 NEXT I
236 L1=POS(G$,".")
237 F$=G$[1,L1-1]
238 F$[L1]=","
239 F$[L1+1]=G$[L1+1]
240 I2=N3/8
245 I3=N3/2
300 REM DEFINE PLOT CONTROL
301 REM P(1),P(2),P(3),P(4) ARE SCALE CONTROL
302 REM PAPER IS 8.5 INCH VERTICAL, 11 INCH WIDE
303 REM P(21)=PAPER HEIGHT/WIDTH
304 REM P(5),P(6),P(7)=BOTTOM,TOP,NO. OF TICS ON VERT. AXIS
305 REM P(8)=HORIZ. POSITION OF VERT. AXIS
306 REM P(9)=SPACING BETWEEN HORIZ. TICS
307 REM P(24),P(18)=CENTER,HALF WIDTH OF HORIZ. AXIS
308 REM P(10),P(11)=H,V COORD. FOR DATE
309 REM P(12),P(13)=H,V FOR TIME
310 REM P(14),P(15)=H,V FOR ALTITUDE
311 REM P(16),P(17)=H,V FOR TITLE
312 REM P(19),P(20)=HEIGHT,ASPECT RATIO OF DATE, TIME, ALT, TITLE
313 REM P(22),P(23)=HORIZ. HALF LENGTH,SPACING OF VERT. TICS
314 REM P(25)=HORIZ. SPACE BETWEEN DATA POINTS
315 REM P(26)=VERT.SCALE FOR DATA
316 REM P(27)=PLOT CONTROL 1=DOT,0=LINES
317 REM P(28)=RELATIVE VERT. OF HORIZ. AXIS LABEL TO AXIS
318 REM P(30),P(31)=H,V FOR SCALING VALUE
319 REM P(29)=SIZE OF AXIS LABEL
320 REM P(32)=V FOR FREQ. AXIS
345 REM READ TABLE, ZERO INTO P(I) WHICH ARE CALCULATED
350 FOR I=1 TO 32
355 READ P[I]
360 NEXT I
370 DATA 0,1100,35,850,200,750,10,0,0,75,775,75,700,75,625
```

Table A.16 cont'd.

```
372 DATA 50,40,400,2,1.7,0,10,0,600,0,0,0,-25,1,60
374 DATA 550,150
400 REM CALCULATE OTHER PLOT PARAMETERS
404 REM P(9)=TIC FOR 1 M/S
405 P[9]=P[18]/14.662
409 REM PUT VERT AXIS AT CENTER OF HORIZ. AXIS
410 P[8]=P[24]
414 REM FIND VERT TIC SPACING
415 P[23]=(P[6]-P[5])/P[7]
419 REM PAPER RATIO USE SCALE STUFF
420 P[21]=(P[4]-P[3])/(P[2]-P[1])
424 REM CALCULATE HORIZ SPACING FOR DATA
425 P[25]=P[18]/I3
429 REM CALCUALTE VERT. MULTIPLIER FOR DATA
430 P[26]=(P[6]-P[5])/999
480 L$="POWER SPECTRUM OF  SECONDS OF COHERENTLY INTEGRATED DATA"
481 U$=T$[1,2]
482 U$[3]=":"
483 U$[4]=T$[3,4]
484 U$[6]=":"
485 U$[7]=T$[5,6]
486 U$[9]=" CST"
490 H5=LEN(H$)
491 H$[H5+1]=" KM"
503 REM START PLOT
505 SCALE P[1],P[2],P[3],P[4]
510 XAXIS P[5],P[9],P[24],P[24]+P[18]
515 XAXIS P[5],-P[9],P[24],P[24]-P[18]
524 REM TIC VERT AXIS
525 FOR I=1 TO P[7]
527 XAXIS P[5]+I*P[23],+1,P[8],P[8]-P[22]
528 XAXIS P[5]+I*P[23],-1,P[8],P[8]+P[22]
529 NEXT I
580 PLOT P[10],P[11],1
581 LABEL (*,P[19],P[20],0,P[21])F$
585 PLOT P[12],P[13],1
586 LABEL (*)U$
590 PLOT P[14],P[15],1
591 LABEL (*)H$
592 PLOT P[30],P[31],1
593 LABEL (594)999/P1
594 FORMAT E8.1
595 PLOT P[16],P[17],1
596 LABEL (*)L$[1,19];I2;L$[20,57]
600 REM PLOT DATA
605 REM LOOP OVER N+1 POINTS
610 FOR I=-I3 TO I3
615 REM CONVERT TO VALUE FROM 0 TO N-1
620 IF I<0 THEN 635
625 D1=I
630 GOTO 640
635 D1=I+N3
640 REM FIND SUBSCRIPTS J,K FOR D(J,K)
645 J=INT(D1/256)+1
650 K=D1+1-256*(J-1)
655 REM FIND HORIZONTAL POSITION
660 X=P[24]+I*P[25]
665 REM FIND VERTICAL POSITION
670 Y=D[J,K]*P[26]+P[5]
675 REM PUT PEN DOWN AFTER MOVING
680 PLOT X,Y,-2
685 REM CHECK PLOT CONTROL FOR DOTS
690 IF P[27]=0 THEN 700
695 PEN
700 NEXT I
1000 REM LABEL HORIZ. AXIS
1001 LABEL (*,P[29],P[20],0,P[21])
1005 Y=P[5]+P[28]
1006 PLOT P[24]+P[18],Y,1
1007 CPLOT 1.5,0
1008 LABEL (*)"M/S"
1010 FOR I=2 TO -2 STEP -1
```

Table A.16 cont'd.


```
1015 I1=5*I
1020 PLOT P[24]+I1*P[9],Y,1
1025 CPLOT F[I+3],0
1030 LABEL (1035)I1
1035 FORMAT F3.0
1040 NEXT I
1060 REM FREQ. AXIS
1065 XAXIS P[32],P[18]/4,P[24]-P[18],P[24]+P[18]
1066 Y=P[32]+P[28]
1067 PLOT P[24]+P[18],Y,1
1068 CPLOT 1.5,0
1069 LABEL (*)"HZ"
1070 FOR I=4 TO -4 STEP -1
1075 PLOT P[24]+I*P[18]/4,Y,1
1080 IF I >= 0 THEN 1095
1085 CPLOT -1,0
1090 GOTO 1100
1095 CPLOT -1.3,0
1100 LABEL (1105)I
1105 FORMAT F2.0
1110 NEXT I
1130 STOP
1135 END
```

the VIA chip on the special interface card which must be in slot 3 of the Apple II. At that point the operator starts the HP-9830 program. This order of operations is made necessary by the handshake limitations of the interface on the HP-9830.

Each variable is read from disk as a string. The length of a string is determined and the string is sent one character at a time. Finally, an ASCII 10 sent to the HP-9830 indicates an end of record. The 8 header variables are sent in lines 132 to 175 and the array is sent in lines 180 to 235. Note that the number of elements in the array is determined in line 53 as the header is read from disk. This allows the program to be used for the various lengths of FFT files. A handshake statement, such as line 150, causes the Apple to loop until the contents of the handshake register change to signal that the HP-9830 is ready for a character. The transfer of the character automatically resets the handshake register. When the array has been transferred the program restarts. The operator has the option to input another file, catalog the disk, or quit. The second file can be loaded while the HP-9830 is plotting the data from the first file. The Apple program will wait at line 155 until the plot is finished, the new paper is placed on the plotter, and the HP-9830 program is restarted.

The HP-9830 program shown in Table A.16 reads in the data from the Apple II and then plots it using axes and labels which are independent of the number of points in the power spectrum to be plotted. Lines 100 to 235 read the header and array variables from the interface one string at a time using the ENTER command. ENTER automatically converts the incoming string of ASCII characters to the form of the specified variable, either numeric or string. The HP-9830 therefore reads a single number from the Apple II using only one BASIC statement. Sending the numbers as bytes would require

disassembly at the Apple II and re-assembly on the HP-9830, a process which would require several BASIC statements. Because of the very slow speed of the HP-9830 BASIC it is desirable to minimize the number of BASIC statements. The numbers are therefore sent as ASCII characters.

The date is stored on Apple disk and sent to the HP-9830 with a dot instead of a comma in order to avoid a problem with commas in Applesoft strings. The comma is inserted in place of the dot in lines 236 to 239. Lines 300 to 400 define and read the plot control matrix. This matrix is used to allow rapid change of the format of the plot during initial runs of the program. In lines 500 to 596 the vertical axis and the plot labels are produced. In order to plot the points of the power spectrum two problems must be addressed. First the array limitation for a single dimension in the HP-9830 is 256 yet the spectrum may be longer. If the array is longer than 256 it is folded into two dimensions as it is read in. The array must then be unfolded as it is plotted. Secondly, the data are read in as all positive frequencies beginning at zero with the first element of the array. The plot shows negative and positive frequencies, however, so the second half of the array must be plotted first. The appropriate array element is determined in lines 615 to 650. The horizontal and vertical plot coordinates are determined in lines 660 and 670 and the point is plotted in line 680. Note that the plot can be a continuous line or a set of dots. (See lines 685 to 695 and line 316.) Finally, horizontal axes for both velocity and frequency are drawn and labeled. The program must be restarted to read and plot the next power spectrum.

A.7 Generation of Random Spectral Files

In this work the spectra obtained when many scatterers are present in the scattering volume are compared to spectra generated from random data.

The program RANDOMFFT8, shown in Table A.17, produces 8-second power spectra files on disk in the same format as the real data. The simulated data are plotted using the programs discussed in the previous section. The FFT algorithm is a BASIC version of the high-speed FORTRAN routine used in the PDP-15 programs to process the real data. The random number generator is based on the routine employed by Horton and Bowhill [1971].

The program RANDOMFFT8 consists of a main body in lines 1 to 1220 followed by subroutines to perform the various array manipulations which are required. The variables in lines 50 to 65 are declared to be integers so that the FOR-NEXT loops of the compiled version RANDOMFFT8.OBJ will run faster. The program calculates a complex series of random numbers equivalent to 128 seconds of data. These data are filtered in the frequency domain with a Gaussian window and transformed back to the time domain. The samples resulting from this series of steps are again transformed to the frequency domain, but at an 8-second spectral length corresponding to the real data. Finally, the 8-second spectrum is squared, normalized and stored to disk.

The routines used to manipulate the complex sample pairs in the arrays RE and IM are shown in lines 7400 to 9935. The coefficient arrays are calculated in lines 7400 to 7490 while the associated FFT routine is in lines 7500 to 7990. Note that one set of coefficients must be calculated for the 1024-point transform and a second set used for the 64-point transform. The square and normalize routine in lines 8300 to 8345 converts the spectrum to the format expected in the disk files. The random number generator in lines 9800 to 9935 is called repeatedly by the routine in lines 9000 to 9055 to generate the random complex samples. The sum of 12 random variables from the generator will have a variance of 1, a mean value of 6, and will be uniformly distributed. The mean value is subtracted to obtain a

Table A.17 RANDOMFFT8.


```
1  REM  PROGRAM FOR SIMULATING SPECTRA
2  REM  WITH GAUSSIAN SHAPE DERIVED FROM
3  REM  RANDOM NUMBERS, THEN TRUNCATING
4  REM  RESULTANT TIME SERIES AND FINDING
5  REM  THE NEW POWER SPECTRUM
10  REM  KEN GIBBS
11  REM  7/20/82
50  REM  ! INTEGER I,J,K,N2,IP,L,L1,LE,LC,N9,N3
55  REM  ! INTEGER NT,KL,KH,KN,TN,N,M,N1,PN
60  REM  ! INTEGER I1,I2
65  REM  ! INTEGER NP,NC,N4
100 D$ =  CHR$ (4)
101 PI = 3.14159265
102 PH = PI / 2
103 P2 = PI * 2
105  DIM RE(1024),IM(1024),CR(512),CI(512)
110  DIM WA(1024),WP(1024)
120  DIM FT(10),FK$(3)
125 C = 997
1000  REM  PUT RANDOM VALUES IN ARRAYS
1001 N = 1024:N2 = N / 2:M = 10:N1 = N - 1
1002  PRINT "FINDING RANDOM VALUES"
1003  INPUT "SEED (0 TO 1) ? ";X
1005  GOSUB 9000
1010  REM  SET UP COEFFECIENTS
1012  PRINT "MAKE COEFFECIENTS FOR FIRST TRANSFORM"
1015  GOSUB 7400
1020  REM  TAKE TRANSFORM
1022  PRINT "TRANSFORM"
1025  GOSUB 7500
1030  REM  MAKE GAUSSIAN WINDOW
1032  PRINT "MAKING GAUSSIAN"
1035  GOSUB 9100
1040  REM  FILTER
1042  PRINT "FILTER"
1045  GOSUB 9200
1050  REM  IDFT
1055  PRINT "INVERSE TRANSFORM"
1060  GOSUB 9300
1065  GOSUB 7500
1070  GOSUB 9300
1075  GOSUB 9400
1080  REM  NOW SET UP THE SHORT FFT
1085 N = 64:N2 = N / 2:M = 6:N1 = N - 1
1090  REM  AND COEFFECIENTS
1092  PRINT "SHORT COEFFECIENTS"
1095  GOSUB 7400
1100  REM  TRANSFORM
1102  PRINT "SHORT TRANSFORM"
1105  GOSUB 7500
1110  REM  SQUARE AND NORMALIZE
1111  REM  TO GET POWER SPECTRUM
1112  PRINT "NORMALIZE"
1115  GOSUB 8300
1120  REM  WRITE TO DISK
1125  PRINT D$;"OPEN RANDOMFFT"
1130  PRINT D$;"DELETE RANDOMFFT"
1135  PRINT D$;"OPEN RANDOMFFT"
1140  PRINT D$;"WRITE RANDOMFFT"
1145  PRINT "NEVER 0. 0000"
1150  PRINT "FFTTEST"
1155  PRINT "000000"
1160  PRINT "0"
1165  PRINT N
1170  PRINT FU
1175  PRINT "0"
1180  PRINT "0"
1190  FOR I = 1 TO N
1195  PRINT RE(I)
1200  NEXT
1205  PRINT D$;"CLOSE RANDOMFFT"
1210  GOSUB 9800
1215  PRINT "SEED FOR NEXT TIME=";X
1220  END
1947  REM  GOSUB 9500
7400  REM  FILL COEFFECIENT ARRAYS
7410 PI = 3.14159265
7420 F1 = PI / N2
7430 ANGLE = 0
7440  FOR I = 1 TO N2
7450 CR(I) =  COS (ANGLE)
7460 CI(I) =  -  SIN (ANGLE)
7470 ANGLE = ANGLE + F1
7480  NEXT
7490  RETURN
7500  REM  COMPLEX FFT,POWER OF 2, IN PLACE
7510  REM  SPECIAL CODE STAGES 1 AND 2
7520  REM  COEFFECIENTS CALCU ATED EXTERNALLY
7530  REM  RE, IM ARRAYS, N=NO. OF POINTS
7540  REM  N2=N/2,N=2**M,N1=N-1
7550  REM  SHUFFLE
7560 J = 1
7570  FOR I = 1 TO N1
7580  IF I >  = J GOTO 7620
7590 TR = RE(J)
7595 TI = IM(J)
7600 RE(J) = RE(I)
7605 IM(J) = IM(I)
7610 RE(I) = TR
7615 IM(I) = TI
7620 K = N2
7630  IF K >  = J GOTO 7670
7640 J = J - K
7650 K = K / 2
7660  GOTO 7630
7670 J = J + K
7680  NEXT
7690  REM  END SHUFFLE, START STAGE 1
7700  FOR I = 1 TO N STEP 2
7710 IP = I + 1
7720 TR = RE(IP)
7725 TI = IM(IP)
7730 RE(IP) = RE(I) - TR
7735 IM(IP) = IM(I) - TI
7740 RE(I) = RE(I) + TR
7745 IM(I) = IM(I) + TI
7750  NEXT
7760  REM  STAGE 2, TWO PARTS
7770  FOR I = 1 TO N STEP 4
7780 IP = I + 2
7790 TR = RE(IP)
7795 TI = IM(IP)
7800 RE(IP) = RE(I) - TR
7805 IM(IP) = IM(I) - TI
7810 RE(I) = RE(I) + TR
7815 IM(I) = IM(I) + TI
7820  NEXT
7830  FOR I = 2 TO N STEP 4
7840 IP = I + 2
7845 TR = IM(IP)
7850 TI =  - RE(IP)
7860 RE(IP) = RE(I) - TR
7865 IM(IP) = IM(I) - TI
7870 RE(I) = RE(I) + TR
7875 IM(I) = IM(I) + TI
7890  NEXT
7900  REM  STAGES 3 TO M
7905 LE = 4
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.17 cont'd.

```
7910  FOR L = 3 TO M
7915 L1 = LE
7920 LE = LE + LE
7925 N9 = N2 / L1
7930 LC = 1
7935  FOR J = 1 TO L1
7940 UR = CR(LC)
7945 UI = CI(LC)
7950 LC = LC + N9
7955  FOR I = J TO N STEP LE
7960 IP = I + L1
7965 TR = RE(IP) * UR - IM(IP) * UI
7970 TI = RE(IP) * UI + IM(IP) * UR
7975 RE(IP) = RE(I) - TR:IM(IP) = IM(I) - TI
7980 RE(I) = RE(I) + TR:IM(I) = IM(I) + TI
7985  NEXT : NEXT : NEXT
7990  RETURN
8100  REM  PRINT OUT RE,IM ARRAYS
8102  REM  TEST ROUTINE
8105  FOR I = 1 TO 128
8110  PRINT I;"  ";RE(I);"  ";IM(I)
8115  NEXT
8120  RETURN
8300  REM  SQUARE,NORMALIZE
8304 MX = 0
8305  FOR I = 1 TO N
8307 RE(I) = RE(I) * RE(I) + IM(I) * IM(I)
8310  IF RE(I) > MX THEN MX = RE(I)
8315  NEXT
8317  IF MX = 0 THEN FU = 1: GOTO 8345
8320 FU = 999 / MX
8325  FOR I = 1 TO N
8330 RE(I) =  INT (FU * RE(I))
8340  NEXT
8345  RETURN
9000  REM  GENERATE RANDOM RE AND IM
9001  REM  ARRAYS FROM A RANDOM AMPLITUDE
9002  REM  AND A RANDOM PHASE
9010  FOR I1 = 1 TO N
9015 R1 = 0:R2 = 0
9020  FOR I2 = 1 TO 12
9022  GOSUB 9800
9025 R1 = R1 + X
9027  GOSUB 9800
9030 R2 = R2 + X
9035  NEXT
9040 R1 = R1 - 6
9045 R2 = R2 - 6
9046 RE(I1) =  SQR (R1 * R1 + R2 * R2)
9047  GOSUB 9800
9048 IM(I1) = P2 * X
9050  NEXT
9051 NC = N
9052  GOSUB 9700
9055  RETURN
9100  REM  CREATE GAUSSIAN FILTER
9105  REM  SC IS A CONSTANT BASED ON
9110  REM  THE DESIRED SPECTRAL WIDTH
9115 SC = N * (N / 16)
9116 PC = 2 * PI / (N * 20)
9120  FOR I = 0 TO N2
9125 WA(I + 1) =  EXP ( - 1 * I * (I / SC))
9126 WP = 0
9129  IF I = 0 THEN 9135
9130 WA(N + 1 - I) = WA(I + 1)
9131 WP(N + 1 - I) =  - WP(I + 1)
9135  NEXT
9140  RETURN
9200  REM  FILTER
9205 NC = N
9210  GOSUB 9600
9215  FOR I = 1 TO N
9220 RE(I) = RE(I) * WA(I)
9225 IM(I) = IM(I) + WP(I)
9230  NEXT
9235  GOSUB 9700
9240  RETURN
9300  REM  CONJUGATE
9305  FOR I = 1 TO N
9310 IM(I) =  - IM(I)
9315  NEXT
9320  RETURN
9400  REM  DIVIDE BY N
9405  FOR I = 1 TO N
9410 RE(I) = RE(I) / N
9415 IM(I) = IM(I) / N
9420  NEXT
9425  RETURN
9500  REM  PRINT RE,IM
9502  PR# 1
9503  PRINT : PRINT
9505  FOR I = 1 TO NP
9510  PRINT I,RE(I),IM(I)
9515  NEXT
9517  PR# 0
9520  RETURN
9600  REM  CONVERT RE,IM TO AMP,PHASE
9601  REM  WITH AMP IN RE,PHASE IN IM
9620  FOR I = 1 TO NC
9625 T1 = RE(I)
9630 T2 = IM(I)
9635 RE(I) =  SQR (T1 * T1 + T2 * T2)
9640  IF T1 = 0 AND T2 = 0 THEN T3 = 0
9645  IF T1 = 0 AND T2 < 0 THEN T3 =  - PH
9650  IF T1 = 0 AND T2 > 0 THEN T3 = PH
9655  IF T1 < 0 THEN T3 = PI +  ATN (T2 / T1)
9660  IF T1 > 0 THEN T3 =  ATN (T2 / T1)
9665  IF T3 > PI THEN T3 = T3 - P2
9670 IM(I) = T3
9675  NEXT
9680  RETURN
9700  REM  CONVERT AMP IN RE AND PHASE
9701  REM  IN IM TO RE AND IM
9705  FOR I = 1 TO NC
9710 T1 = RE(I)
9715 T2 = IM(I)
9720 RE(I) = T1 *  COS (T2)
9725 IM(I) = T1 *  SIN (T2)
9730  NEXT
9735  RETURN
9800  REM  RANDU ROUTINE
9805 X = X * C
9810 X = X -  INT (X)
9815  RETURN
9905  FOR I = 1 TO N
9910 RE(I) = 0
9915 IM(I) = 0
9920  NEXT
9925 RE(1) = 10
9935  RETURN
```

0 mean distribution. Two values are used as real and imaginary components and the amplitude is then calculated. The phase is also randomly assigned. In practice the real and imaginary values can be used as the final random sample without any effect on the results. The Gaussian filter created in lines 9100 to 9140 is very wide to simulate the wide spectra found when many scatterers are present. The filter is centered about zero frequency because the Doppler frequency shift should have no effect on the results. Furthermore, the filter introduces no phase shift in order to maintain the random phase of the time sequence. The filtering operation in lines 9200 to 9240 consists of multiplying the output of the transform of the random samples by the filter values. The remainder of the program consists of routines to convert between the rectangular and polar forms of the complex samples and to complete the inverse FFT. The RANDOMFFT8.OBJ program requires about 2 minutes to produce a single 8-second random spectra on disk.

A.8 <u>Doppler-Aided Altitude Determination</u>

The plots of scattering layers using increased altitude resolution are the result of the three programs discussed in this section. The ALTITUDE FINDER program, shown in Table A.18, is the program initially used to find the altitudes from printouts of the power spectra. These printouts were obtained by routing the output of the PDP-15 programs which generate the FFTs, shown above in Tables A.6 and A.13, to the printing terminal instead of to the video terminal for transfer to the Apple II. ALTITUDE FINDER uses the scaling factors and the value of a given component for the spectra at 5 adjacent altitudes. The altitude for that component, based on a parabolic fit to the variation in amplitude, is determined and printed. The altitude for several components can be determined without reentering the scaling factors. The base height input required is the altitude 1.5 km below the

Table A.18 ALTITUDE FINDER.


```
10  DIM F(20),N(20)
12  INPUT "BASE HEIGHT ? ";BH
15  INPUT "START HEIGHT NO. ? ";HS
20  IF HS < 1 OR HS > 20 GOTO 15
25  FOR HG = HS TO HS + 4
30  PRINT "HEIGHT ";HG;" FUDGE "
35  INPUT F(HG)
40  NEXT
45  INPUT "COMPONENT NUMBER ? ";CN
50  IF  ABS (CN) > 32 THEN 1000
55  FOR HG = HS TO HS + 4
60  PRINT "HEIGHT ";HG;" VALUE"
65  INPUT N(HG)
70  NEXT
75  GOSUB 500
80  GOTO 45
500 MX = 0
502 NM = 0
510  FOR HG = HS TO HS + 4
515 N(HG) =  SQR (N(HG) / F(HG))
520  NEXT
525  FOR HG = HS + 1 TO HS + 3
530  IF N(HG) > MX THEN MX = N(HG):NM = HG
535  NEXT
540  IF NM > 0 GOTO 600
550  PRINT "NO HEIGHT FOUND FOR ";CN
560  RETURN
600 T = 5 * (N(NM - 1) - N(NM + 1)) / (N(NM - 1) + N(NM + 1) - 2 * N(NM))

610 AL = BH + 1.5 * NM + 1.5 * T / 10
620  PRINT "COMPONENT ";CN;" -> ";AL;" KM"
630  RETURN
1000  END
```

altitude of the first sample. The start height input is the number of the lowest height in the group of 5 heights which will span the scattering region. The program expects the scaling factors and component values in order beginning with the lowest altitude of the five. The program exits when a component value of magnitude greater than 32 is entered.

The CONTOUR program, shown in Table A.19, aids in the selection of the appropriate spectral components to plot by printing a table of power versus frequency component as a function of time. The program searches a floppy disk for the desired power spectrum, calculates the power in dB above 50 dB for frequency components numbered -15 to 15 and prints the results at evenly spaced intervals across the page using the escape-code tab control of the Spinwriter. The program takes a starting time and altitude as input to determine the first spectrum to read from disk. The time portion of the filename is incremented and the printer paper is advanced to print the output from the next spectra on the following line. The program loops until it is unable to locate the necessary file on the floppy disk. Recall that the files are stored on disk during transfer by calculating all spectra for a given time. Therefore, the spectra for a given altitude may be distributed over several disks. The missing file is indicated and the operator is prompted to try again or quit. If another disk with the missing file is inserted the program will continue generating the contour table. Contours are hand-drawn around the numbers on the printed table.

The program FFT8ALTITUDES, shown in Table A.20, is used to automate the process of determining the altitude of a particular frequency component. As in the program ALTITUDE FINDER, the value of a given component at five consecutive altitudes is needed to determine the precise altitude. FFT8ALTITUDES reads the entire spectrum from disk for the 5 altitudes and

Table A.19 CONTOUR.

```
1   REM  PROGRAM TO PRINT THE TABLE
2   REM  FROM WHICH A CONTOUR MAP
3   REM  OF SPECTRAL COMPONENT VS TIME
4   REM  IS MADE---PRINTS THE VALUE IN
5   REM  DB ABOVE 50 DB
6   REM  KEN GIBBS
7   REM  7/5/82
10  REM  ! INTEGER I,K
50  DIM A(64),G(31)
100 D$ =  CHR$ (4)
101 TB$ =  CHR$ (9):ES$ =  CHR$ (27)
105 BS = 50
110 LT =  LOG (10)
1100  PRINT "8 SECOND FFT CONTOUR PLOT"
1105  INPUT "ALTITUDE IN KM? ";HT
1106  IF  INT (HT / 1.5) <  > HT / 1.5 GOTO 1105
1110  INPUT "TIME (E.G. 093800)? ";T$
1115  IF  LEN (T$) <  > 6 GOTO 1110
1120 HR =  VAL ( LEFT$ (T$,2))
1125 MN =  VAL ( MID$ (T$,3,2))
1130 SC =  VAL ( MID$ (T$,5,2))
1500  REM  FILE NAME TO READ NEXT
1505 F$ = "FFT8/" + T$ + "/" +  STR$ (HT)
1510  ONERR  GOTO 2000
1515  REM  CHECK FOR FILE
1520  PRINT D$;"VERIFY ";F$
1525  PRINT D$;"OPEN ";F$
1530  PRINT D$;"READ ";F$
1532  PRINT "READING ";F$
1535  REM  DON'T NEED THE FIRST FOUR
1540  FOR I = 1 TO 4: INPUT TS$: NEXT
1545  INPUT N3: INPUT FU
1550  REM  WASTE TWO MORE
1555  INPUT TS$: INPUT TS$
1560  REM  NOW GET THE FFT
1565  FOR I = 1 TO N3
1570  INPUT A(I)
1575  NEXT
1580  PRINT D$;"CLOSE ";F$
1600  REM  CALCULATE THE OUTPUT ARRAY
1605  REM  TAKE ONLY -15 TO +15 COMPONENTS
1610  FOR I =  - 15 TO 15
1615  IF I < 0 THEN J = I + 65
1620  IF I >  = 0 THEN J = I + 1
1625 K = I + 16
1630 G(K) =  INT (10 *  LOG ((A(J) + .5) / FU) / LT + .5) - BS
1635  NEXT
1640  REM  NOW READY TO PRINT ANOTHER ROW
1650  GOSUB 3000
1655  REM  INCREMENT TIME AND GO AGAIN
1660 SC = SC + 10
1665  IF SC < 60 GOTO 1685
1670 SC = 0:MN = MN + 1
1675  IF MN < 60 GOTO 1685
1680 MN = 0:HR = HR + 1
1685 T$ =  RIGHT$ ( STR$ (100 + HR),2) +  RIGHT$ ( STR$ (100 + MN),2) +  RIGHT$ ( STR$ (100 + SC),2)
1690  GOTO 1500
2000  REM  CAN'T FIND FILE
2005  POKE 216,0
2010  PRINT "CAN'T FIND ";F$
2015  PRINT "T)RY AGAIN OR Q)UIT"
2020  INPUT "INPUT T OR Q ";Q$
2025  IF  LEFT$ (Q$,1) = "T" GOTO 1500
2030  IF  LEFT$ (Q$,1) = "Q" GOTO 2500
2035  GOTO 2015
2500  REM  QUIT ROUTINE
2505  PRINT "THAT'S ALL"
2510  END
3000  REM  PRINT ONE LINE HERE
3005  PRINT D$;"PR#1"
3010  PRINT T$;
3015  FOR I = 1 TO 31
3020 TP = 6 + I * 3
3025  PRINT ES$;TB$; CHR$ (TP);G(I);
3030  NEXT
3035  PRINT : PRINT
3040  PRINT D$;"PR#0"
3045  RETURN
```

Table A.20 FFT8ALTITUDES.


```
100  DIM A(5,65)
105 D$ =  CHR$ (4)
110  DIM CV$(20)
115  DIM AS$(5)
1500  REM  GET THE VOLUME INFO AND MAKE STIRNGS
1505  INPUT "START VOLUME? ";SV
1510  IF SV < 10 OR SV > 18 GOTO 1505
1515  INPUT "END VOLUME? ";EV
1520  IF EV < 11 OR EV > 18 GOTO 1515
1525  FOR I = SV TO EV
1530 CV$(I) = ",S6,D1,V" +  STR$ (I)
1535  NEXT
1540 VN = SV
1541 GV = SV
2000  REM  GET ALTITUDE INFO
2005  INPUT "LOW ALTITUDE? ";LA
2010  IF  INT (LA / 1.5) <  > LA / 1.5 GOTO 2005
2015  REM  MAKE ALTITUDE STRINGS
2020  FOR I = 1 TO 5
2025 AS$(I) = "/" +  STR$ ((I - 1) * 1.5 + LA)
2030  NEXT
2500  REM  GET COMPONENT RANGE
2505  INPUT "LOWEST COMPONENT NO.? ";LC
2510  IF LC <  - 32 OR LC > 32 GOTO 2505
2515  INPUT "HIGHEST COMPONENT NO.?";HC
2520  IF HC > 32 OR HC < LC GOTO 2515
2525  REM  SHIFT FOR ARRAY REFERENCE
2530 LC = LC + 33:HC = HC + 33
2600  REM  START,END TIMES
2605  INPUT "START TIME: HR,MN,SC ";SH,SM,SS
2607  IF SH < 0 OR SM < 0 OR SS < 0 GOTO 2605
2610  INPUT "END TIME:HR,MN,SC ";EH,EM,ES
2615  IF EH < SH GOTO 2610
2620  IF EH = SH AND EM < SM GOTO 2610
2625  IF EH = SH AND EM = SM AND ES < SS GOTO 2610
2690  REM  MAKE TIME STRING
2695 T$ =  RIGHT$ ( STR$ (100 + SH),2) +  RIGHT$ ( STR$ (100 + SM),2) +  RIGHT$ ( STR$ (100 + SS),2)
2700  REM  LOOP OVER ALTITUDE
2705 J = 1
2710 F$ = "FFT8/" + T$ + AS$(J)
2725  ONERR  GOTO 7000
2727 ER = 0
2730  PRINT D$;"VERIFY ";F$;CV$(VN)
2731  IF ER = 1 GOTO 2727
2732 GV = VN
2735  PRINT D$;"OPEN ";F$
2740  PRINT D$;"READ ";F$
2745  PRINT "READING ";F$
2750  ONERR  GOTO 8000
3000  REM  GET FFT IN,FOLD ARRAY
3005  REM  COMPONENT 0 INTO ELEMENT 33
3010  REM  WASTE THE FIRST FIVE
3015  FOR I = 1 TO 5: INPUT TS$: NEXT
3020  INPUT FU
3025  INPUT TS$: INPUT TS$: REM  WASTE TWO MORE
3030  REM  GET ARRAY
3035  FOR I = 33 TO 65
3040  INPUT A(J,I)
3045  NEXT
3050  FOR I = 2 TO 32
3055  INPUT A(J,I)
3060  NEXT
3065 A(J,1) = A(J,65)
3070  REM  UNNORMALIZE
3075  FOR I = 1 TO 65
3080 A(J,I) =  SQR (A(J,I) / FU)
3085  NEXT
3090  PRINT D$;"CLOSE ";F$
3095 J = J + 1
```

Table A.20 cont'd.

ORIGINAL PAGE IS
OF POOR QUALITY

```
3100  IF J < = 5 GOTO 2710
3105  REM  GOT ALL FIVE FFTS IN
3185  POKE 216,0
3190  PRINT D$;"PR#1"
3195  PRINT T$
3197 CM = 1
3200  FOR I = LC TO HC
3215 MX = 0:NM = 0
3220  FOR J = 2 TO 4
3225  IF A(J,I) > MX THEN MX = A(J,I):NM = J
3230  NEXT
3235  IF NM > 0 GOTO 3246
3240  PRINT "NO HEIGHT FOUND FOR ";I
3245  GOTO 3275
3246 T1 = A(NM - 1,I):T2 = A(NM + 1,I)
3250 T = 5 * (T1 - T2) / (T1 + T2 - 2 * A(NM,I))
3255 AL = .15 * T + (NM - 1) * 1.5 + LA
3260  PRINT I - 33,AL,
3265  IF CM = 2 THEN CM = 1: PRINT : GOTO 3275
3270  IF CM = 1 THEN CM = 2
3275  NEXT
3280  PRINT : PRINT
3290  PRINT D$;"PR#0"
3300  REM  MAKE NEW TIME
3305 SS = SS + 10
3310  IF SS < 60 GOTO 3330
3315 SS = SS - 60:SM = SM + 1
3320  IF SM < 60 GOTO 3330
3325 SM = SM - 60:SH = SH + 1
3330  IF EH < SH GOTO 3350
3335  IF EH = SH AND EM < SM GOTO 3350
3340  IF EH = SH AND EM = SM AND ES < SS GOTO 3350
3345  GOTO 2690
3350  REM  QUIT
3355  END
7000  REM  COULDN'T FIND FILE
7005 VN = VN + 1
7010  IF VN > EV THEN VN = SV
7015  IF VN = GV GOTO 7030
7017 ER = 1
7020  RESUME
7030  PRINT "CAN'T FIND ";F$
7035  STOP
8000  REM  ERROR WHILE READING
8005  PRINT D$;"CLOSE"
8010  POKE 216,0
8015  PRINT "ERROR WHILE READING FILE"
8020  STOP
```

performs the calculation for a range of components. The program also loops over time. This requires that a large number of spectra be available. The solution is to transfer all of the necessary spectra for a given run to the Corvus hard disk in contiguous volumes. The program sca[illegible]l volumes for a given file to permit unattended operation. The presence of a file is determined by the VERIFY command of Apple DOS and the absence is handled with ONERR GOTO commands. Therefore, the loops in the program cannot be FOR-NEXT loops. The program will search all specified volumes for a given file, however, it should be noted that repeated use of the ONERR GOTO statement of Applesoft causes the processor stack to overflow and the system to hang. In general the files are stored in chronological order on the disks as they are transferred from the PDP-15. The floppy disks should be transferred to the Corvus with the earlier spectra in the lowest volume number of the group used for this program. This order minimizes the number of times that the VERIFY command fails. The output consists of two columns indicating the component number and corresponding altitude.

A.9 Generation of Files of Hourly Statistics

The program DGEN3, shown in Table A.21, calculates the hourly statistics for the minute-by-minute power and velocity data and stores them on disk. The program is designed to process all of the power and velocity files on a given data disk and produce a second disk containing the hourly statistics. Like the data disks, the processed disks have a date file and are stored with one day per disk. DGEN3 contains improved versions of the calculation routines used in the DVEL PLOT and DPOW PLOT programs discussed by Roth [1982].

The series of routines in lines 6000 to 6930 are for input of the slot, drive and volume of the data or source disk and the destination or processed

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.21 DGEN3.

```
1 REM CALCULATE HOURLY STATISTICS
2 REM OF CS DATA ON APPLE DISK
5 REM RESULTS FOR EACH HEIGHT ON DISK
8 REM SOURCE,DESTINATION MUST BE AT DIFFERENT S,D,V
10 REM KEN GIBBS
11 REM 6/16/82
50 REM ! INTEGER I,J,RP,PF,I1,I3,HR,C,CA,CV,CP,MO,CW,CM,CI
55 REM ! INTEGER NT,KL,KH,KN,TN,A1,A2,A3,A4,C5,C9,IH
60 REM ! INTEGER K3,K4,V2,V1,I6,J1
65 REM ! INTEGER N4(2),TM,A5,LP,A7
100 D$ = CHR$ (4)
105 DIM G(60),MN(2),E(20,12)
120 DIM FT(10),FK$(3)
122 DIM A(20,120)
145 REM INIT S,D,V TO ALLOW SOURCE<>DEST TEST
150 SS = 0:SD = 1:SV = 0
155 DS = 0:DD = 1:DV = 0
1000 REM SET RP=4 TO WEAVE THROUGH THE DATE AND S,D,V ROUTINES
1005 RP = 4
1010 GOTO 6600
3515 GOSUB 7700: REM INTREE
4000 REM CHECK DISKS
4020 ONERR GOTO 6300
4030 PRINT D$;"VERIFY ";H$;",S"; STR$ (SS);",D"; STR$ (SD);",V"; STR$ (S
V)
4040 ONERR GOTO 6200
4050 PRINT D$;"VERIFY ";H$;",S"; STR$ (DS);",D"; STR$ (DD);",V"; STR$ (D
V)
4060 ONERR GOTO 6100
4070 PRINT D$;"OPEN DUMMY"
4080 PRINT D$;"DELETE DUMMY"
5000 REM GET FILE INFO.
5010 REM ZERO FOR TIME INPUT TERMINATES
5020 REM PROGRAM WILL SKIP BAD FILE NAMES
5030 REM SO DON'T WORRY ABOUT ERRORS
5100 NT = 0
5110 INPUT "TIME (E.G. 1029) ? ";TM
5115 IF TM = 0 AND NT = 0 GOTO 5740
5120 IF TM = 0 AND NT < > 0 GOTO 5155
5125 NT = NT + 1
5130 FT(NT) = TM
5135 IF NT < 10 GOTO 5110
5155 KL = 1:KH = 0
5160 INPUT "POWER FILES? Y OR N ";Q$
5170 IF LEFT$ (Q$,1) = "Y" THEN KH = KH + 1:FK$(KH) = "POWW": GOTO 5200

5180 IF LEFT$ (Q$,1) = "N" GOTO 5200
5190 GOTO 5160
5200 INPUT "VELOCITY FILES? Y OR N ";Q$
5210 IF LEFT$ (Q$,1) = "Y" THEN KH = KH + 1:FK$(KH) = "VELL": GOTO 5240

5220 IF LEFT$ (Q$,1) = "N" GOTO 5240
5230 GOTO 5200
5240 IF KL > KH GOTO 5740
5300 TN = 1
5310 KN = KL
5320 F$ = FK$(KN) + "/" + STR$ (FT(TN))
5324 REM SET THE POWER FILE FLAG
5325 IF LEFT$ (FK$(KN),1) = "P" THEN PF = 1
5326 IF LEFT$ (FK$(KN),1) < > "P" THEN PF = 0
5330 ONERR GOTO 5660
5340 PRINT D$;"VERIFY ";F$;",S"; STR$ (SS);",D"; STR$ (SD);",V"; STR$ (S
V)
5345 PRINT "READING ";F$
5350 PRINT D$;"OPEN ";F$
5360 PRINT D$;"READ ";F$
5370 INPUT L$: INPUT H3: INPUT M3: INPUT N3
5380 INPUT S2: INPUT L2: INPUT A6: INPUT H2
5385 REM HR=NO. OF HOURS
```

Table A.21 cont'd.

```
5386  REM  N4(I)=NO. OF MINUTES IN HOUR (I)
5387  IF N3 <  = 0 OR N3 > 120 GOTO 5660
5388  IF N3 > 0 AND N3 <  = 60 THEN N4(1) = N3:HR = 1: GOTO 5400
5389  REM  N3>60 AND N3<=120 HERE
5390 N4(1) = 60
5391 N4(2) = N3 - 60
5392 HR = 2
5400  FOR I = 1 TO 20
5410  FOR J = 1 TO 120
5420  INPUT A(I,J)
5430  IF A(I,J) > 8000 THEN A(I,J) = 0
5440  NEXT
5450  NEXT
5460  PRINT D$;"CLOSE ";F$
5470  REM  DONE WITH INPUT, START OUTPUT WITH HEADER
5475  PRINT "DONE READING ";F$
5480 F1$ = "D" +  LEFT$ (FK$(KN),3) + "/" +  STR$ (FT(TN))
5485  PRINT F1$
5490  ONERR  GOTO 6030
5500  PRINT D$;"OPEN ";F1$;",S"; STR$ (DS);",D"; STR$ (DD);",V"; STR$ (DV
     )
5510  PRINT D$;"DELETE ";F1$
5515  ONERR  GOTO 6000
5520  PRINT D$;"OPEN ";F1$
5530  PRINT D$;"WRITE ";F1$
5535  REM  CHANGE L$
5540 L$ = H$ + " " +  LEFT$ (F1$,4)
5545  PRINT L$: PRINT H3: PRINT M3: PRINT N3
5550  PRINT S2: PRINT L2: PRINT A6: PRINT H2
5555  PRINT D$;"CLOSE ";F1$
5560  POKE 216,0
5565  IF PF = 1 THEN  GOSUB 7400: REM  POWER CALC
5570  IF PF <  > 1 THEN  GOSUB 7000: REM  VEL CALC
5610  ONERR  GOTO 6000
5615  PRINT D$;"APPEND ";F1$
5617  PRINT D$;"WRITE ";F1$
5619  REM  SET COLUMN COUNT,PRINT SPECIAL POWER STUFF
5620  IF PF = 1 THEN  PRINT MN(1): PRINT MN(2):CM = 4
5625  IF PF = 0 THEN CM = 12
5629  REM  NOW PRINT E TABLE
5630  FOR IH = 1 TO 20
5635  FOR I = 1 TO CM
5640  PRINT E(IH,I)
5645  NEXT
5650  NEXT
5660  PRINT D$;"CLOSE ": REM  CLOSES INPUT FILE ONERR, OUTPUT NORMALLY
5670  POKE 216,0
5680 KN = KN + 1
5690  IF KN <  = KH GOTO 5320: REM  SAME TIME,NEW KIND
5700  REM  DONE ALL FILES WITH THIS TIME
5710 TN = TN + 1
5720  IF TN <  = NT GOTO 5310: REM  NEW TIME
5730  REM  DONE ALL FILES
5740  PRINT "& AND RETURN TO RESTART"
5750  END
6000  REM  DESTINATION DISK FULL
6010  REM  S,D,V SET TO DESTINATION
6015  PRINT D$;"CLOSE ";F1$
6020  PRINT D$;"DELETE ";F1$
6030  REM  THIS ENTRY FOR UNABLE TO OPEN
6040  PRINT "NO ROOM FOR FILE ";F1$
6050  PRINT "FIX A NEW DISK AND THEN RUN"
6060  PRINT "THIS PROGRAM AGAIN WITH THE"
6070  PRINT "WITH THE REVISED LIST OF FILES"
6080  GOTO 5740
6100  REM  DESTINATION IS WRITE PROTECTED
6110  PRINT "CAN'T WRITE TO DESTINATION"
6120 RP = 1: REM  RETURN POINTER TO 4060
6130  GOTO 6800
```

Table A.21 cont'd.

```
6200  REM  WRONG DESTINATION
6210  PRINT H$;" IS NOT AT DESTINATION S,D,V"
6220 RP = 2: REM  RETURN POINTER TO 4040
6230  GOTO 6800
6300  REM  WRONG SOURCE
6310  PRINT H$;" IS NOT AT SOURCE S,D,V"
6320 RP = 3: REM  RETURN POINTER TO 4020
6330  GOTO 6800
6400  REM  SOURCE S,D,V INPUT
6410  INPUT "SOURCE S,D,V ? ";SS,SD,SV
6420  IF SS > 6 OR SS < 4 OR SD < 1 OR SD > 2 OR SV < 0 GOTO 6410
6425  IF SS = DS AND SD = DD AND SV = DV THEN  GOSUB 8000: GOTO 6410
6430  IF RP = 4 GOTO 6500
6435 RP = 3: REM  FORCE CHECKS
6440  GOTO 6800
6500  REM  DESTINATION S,D,V INPUT
6510  INPUT "DESTINATION S,D,V ? ";DS,DD,DV
6520  IF DS < 1 OR DS > 6 OR DD < 1 OR DD > 2 OR DV < 0 THEN 6500
6525  IF SS = DS AND SD = DD AND SV = DV THEN  GOSUB 8000: GOTO 6510
6530  IF RP = 4 GOTO 4000
6540 RP = 2: REM  FORCE DEST. CHECK
6550  GOTO 6800
6600  REM  GET DATE
6610  INPUT "DATE (E.G. MAY 14. 1980)? ";H$
6620  IF RP = 4 GOTO 6400
6625 RP = 3: REM  FORCE CHECKS FOR DATE CHANGE
6630  GOTO 6800
6800  REM  HANDLE DISK SET UP ERRORS
6810  PRINT "DO YOU WANT TO: "
6820  PRINT "  1. CHANGE DATE"
6830  PRINT "  2. CHANGE SOURCE S,D,V"
6840  PRINT "  3. CHANGE DEST. S,D,V"
6850  PRINT "  4. NO CHANGE, TRY AGAIN"
6860  PRINT "  5. QUIT"
6870  INPUT "INPUT 1 TO 5 ";Q
6880  IF Q < 1 OR Q > 5 THEN  PRINT : GOTO 6800
6890  ON Q GOTO 6600,6400,6500,6900,5740
6900  REM  NOW REENTER MAIN FLOW
6910  ON RP GOTO 4060,4040,4020
6920  PRINT : PRINT "ERROR IN RP CONTROL"
6930  STOP
7000  REM  HANDLE VELOCITY
7010  REM  RESTORE VEL CONSTANTS TO
7011  REM  TO AVOID INTERACTION WITH POWER
7015 W1 = 0.5:T9 = 0.6643:T7 = 38.202:T8 = 1:T6 = T7 / 100
7050  FOR I1 = 1 TO HR
7051 CA = (I1 - 1) * 6 + 1: REM  AVERAGE COLUMN POINTER
7052 CV = CA + 1: REM  VARIANCE COLUMN POINTER
7053 CP = CA + 2: REM  NO. POINTS COLUMN POINTER
7054 MO = (I1 - 1) * 60: REM  MINUTE OFFSET
7055 CW = CA + 3: REM  CORRELATION TIME COLUMN POINTER
7056 CI = CA + 4: REM  INDEPENDENT POINT COLUMN POINTER
7057 CM = CA + 5: REM  VARIANCE OF MEAN COLUMN POINTER
7060  FOR IH = 20 TO 1 STEP  - 1
7061 AV = 0:VR = 0:C = 0
7065  FOR I3 = 1 TO N4(I1)
7070 V = A(IH,MO + I3)
7075  IF V = 0 GOTO 7100
7080 AV = AV + V
7085 VR = VR + V * V
7090 C = C + 1
7100 G(I3) = V
7105  NEXT
7110 E(IH,CP) = C
7114  REM  IF C=0 THEN VR AND AV ARE STILL 0
7115  IF C = 0 GOTO 7135
7120 AV = AV / C
7125 VR = VR / C - AV * AV
7130  IF VR < 0 THEN VR = 0
```

Table A.21 cont'd.

```
7135 E(IH,CA) = AV
7140 E(IH,CV) = VR
7144  REM  DEFAULT ENTRY TO ACF FOR C<=3
7145  IF E(IH,CP) <  = 3 THEN  GOSUB 7885: GOTO 7155
7150  GOSUB 7800
7155 E(IH,CI) =  ABS (E(IH,CP) * T9 / E(IH,CW))
7160  IF E(IH,CI) > E(IH,CP) THEN E(IH,CI) = E(IH,CP)
7165  IF E(IH,CI) < 2 AND IH = 20 THEN E(IH,CV) =  - T8: GOTO 7180
7170  IF E(IH,CI) < 2 THEN E(IH,CV) =  -  ABS (E(IH + 1,CV)): GOTO 7180
7175 E(IH,CV) =  SQR (E(IH,CV) * E(IH,CI) / (E(IH,CI) - 1)) / 100
7180  IF E(IH,CP) = 0 GOTO 7195
7185 E(IH,CM) = T7 * E(IH,CV) /  SQR (E(IH,CI))
7190 E(IH,CA) = T6 * E(IH,CA)
7195  NEXT : REM  HEIGHT=IH LOOP
7200  NEXT : REM  HOUR=I1 LOOP
7204  REM  ZERO SECOND HOUR?
7205  IF HR = 2 GOTO 7235
7210  FOR IH = 1 TO 20
7215  FOR I = 7 TO 12
7220 E(IH,I) = 0
7225  NEXT
7230  NEXT
7235  RETURN
7400  REM  HANDLE POWER FILES
7402 A3 = 1:A4 = 0: REM  SORT CONTROL
7405  FOR I1 = 1 TO HR
7406 A1 = N4(I1)
7407  IF  INT (A1 / 2) = A1 / 2 THEN A2 = A1 / 2 + 1
7408  IF  INT (A1 / 2) <  > A1 / 2 THEN A2 = (A1 + 1) / 2
7409 A5 = A1 - A2
7410 MO = (I1 - 1) * 60
7411 C5 = 2 * I1 - 1:C9 = C5 + 1
7412 MN(I1) = 9999
7415  FOR IH = 1 TO 20
7421  FOR I = 1 TO A1
7422 G(I) = A(IH,MO + I)
7423  NEXT
7424  IF A1 = 1 GOTO 7426: REM  SKIP SORT
7425  GOSUB 7500: REM  SORT
7426 LP = 50: GOSUB 8100
7427 E(IH,C5) = AW
7428 LP = 90: GOSUB 8100
7429 E(IH,C9) = AW
7437  REM  SORT THE REST FOR MIN
7438  FOR I = 1 TO A5
7439  IF G(I) < MN(I1) THEN MN(I1) = G(I)
7440  NEXT
7441  NEXT
7442  REM  SUBTRACT HOUR MIN
7443  FOR IH = 1 TO 20
7445 E(IH,C5) = E(IH,C5) - MN(I1)
7446 E(IH,C9) = E(IH,C9) - MN(I1)
7447  NEXT
7448  NEXT
7449  REM  ZERO SECOND HOUR?
7450  IF HR = 2 GOTO 7455
7451  FOR IH = 1 TO 20
7452 E(IH,3) = 0:E(IH,4) = 0
7453  NEXT
7454 MN(2) = 0
7455  RETURN
7500  REM  SORT BASED ON FLOYD'S TREE-NLOG(2)N COMPARISONS
7501  REM  A1 IS LENGTH OF ARRAY G TO BE SORTED
7502  REM  A2=NO. OF ELEMENTS IN ORDER AT TOP OR BOTTOM
7503  REM  A3=1 FOR MAX ON EACH PASS,A3=0 FOR MIN
7504  REM  SORTED VALUES ARE MOVED TOWARD G(A1) WITH A4=0
7505  REM  SORTED VALUES ARE FLIPPED AFTER SORT WITH A4=1
7510  FOR I = 2 TO A1
7515  GOSUB 7700: REM  INTREE
```

Table A.21 cont'd.

```
7520  NEXT
7525  FOR I = A1 TO 2 STEP  - 1
7530 T9 = G(1)
7535 G(1) = G(I)
7540 G(I) = T9
7545  IF I = A1 + 1 - A2 GOTO 7560: REM  DONE
7550  GOSUB 7600: REM  OUTTREE
7555  NEXT
7560  REM  FLIP IF NEEDED
7565  IF A4 = 0 GOTO 7595
7570  FOR I = 1 TO  INT ((A1 + 1) / 2)
7575 T9 = G(I)
7580 G(I) = G(A1 + 1 - I)
7585 G(A1 + 1 - I) = T9
7590  NEXT
7595  RETURN
7600  REM  OUTTREE
7605 K3 = 1
7610 T9 = G(1)
7615 K4 = K3 + K3
7620  IF K4 > I - 1 THEN G(K3) = T9: RETURN
7625  IF K4 + 1 > I - 1 GOTO 7635
7630  IF G(K4 + 1) > G(K4) AND A3 = 1 OR G(K4 + 1) < G(K4) AND A3 = 0 THEN
     K4 = K4 + 1
7635  IF G(K4) <  = T9 AND A3 = 1 OR G(K4) >  = T9 AND A3 = 0 THEN G(K3) =
     T9: RETURN
7640 G(K3) = G(K4)
7645 K3 = K4
7650  GOTO 7615
7700  REM  INTREE
7705 T9 = G(I)
7710 K3 = I
7715  IF K3 <  = 1 THEN G(K3) = T9: RETURN
7720 K4 =  INT (K3 / 2)
7725  IF T9 <  = G(K4) AND A3 = 1 OR T9 >  = G(K4) AND A3 = 0 THEN G(K3) =
     T9: RETURN
7730 G(K3) = G(K4)
7735 K3 = K4
7740  GOTO 7715
7800  REM  ACF
7802 U2 = E(IH,CA) * 2
7805 FL = E(IH,CV)
7807 V2 = 1
7810 TV = W1 * FL
7812 GL = FL
7815  FOR J = 2 TO 13
7816 J1 = J - 1
7817 LL = 0
7818 V1 = 0
7820  FOR I = 1 TO N4(I1) - J1
7822 I6 = I + J1
7825  IF G(I) = 0 OR G(I6) = 0 GOTO 7840
7830 LL = LL + G(I) * G(I6)
7835 V1 = V1 + 1
7840  NEXT
7845  IF V1 = 0 GOTO 7885
7850  IF V1 < 3 GOTO 7870
7855 LL = LL / V1 - U2
7860  IF LL <  = TV GOTO 7900
7865 V2 = J
7867 GL = LL
7870  NEXT
7875 E(IH,CW) = 12
7880  RETURN
7885  IF IH = 20 THEN E(IH,CW) =  - T9: RETURN
7890 E(IH,CW) =  -  ABS (E(IH + 1,CW))
7895  RETURN
7900 E(IH,CW) = V2 - 1 + (J - V2) * (GL - TV) / (GL - LL)
7905  RETURN
8000  REM  SOURCE AND DEST. DIFFERENT
8005  REM  TO MAKE LIFE SIMPLE
8010  PRINT "CAN'T HAVE SAME S,D,V"
8015  PRINT "FOR SOURCE AND DEST."
8020  RETURN
8100  REM  GIVEN LP PERCENT, THIS ROUTINE
8105  REM  FINDS THE LP PERCENTILE
8110  REM  OF THE G ARRAY
8115  REM  ANSWER RETURNED IN AW
8130 A8 = LP * A1 / 100
8135 A7 =  INT (A8)
8140  IF A8 = A7 THEN AW = (G(A7) + G(A7 + 1)) / 2
8145  IF A8 <  > A7 THEN AW = G(A7 + 1)
8150  RETURN
```

disk and the date of the data. As indicated in line 1000, the original call to these routines is made with the control variable RP set to 4. This forces execution of all the routines. The drive parameters for the source and destination volumes can not be the same. The disks are checked for the correct date and a parameter modification loop is used to change the date or the disk parameters if an error occurs. Alternate values of the RP variable are used to control the use of the input routines and to force the disk checks until the date and the disks match.

Each floppy disk of minute-by-minute data contains files of power, velocity, and correlation for each file time. Recall that a single two-hour tape of autocovariance data collected by the PDP-15 is processed three times, once for each type of data. DGEN3 can process the power files, the velocity files or both. DGEN3 can process the power files, the velocity files or both for a list of file times. The routine in lines 5000 to 5135 reads a list of file times until a time of 0 is input. Up to 10 times may be entered. The type of files to be processed is entered in lines 5155 to 5240.

The main program loops are the file-time loop from line 5300 to line 5720 and the file-type loop from line 5310 to line 5690. Each pass through the inner loop causes a file to be read from the data disk, processed, and the results stored on the processed disk. The header is read in lines 5345 to 5380 so that the number of minutes in the file can be used in lines 5385 to 5392 to determine the number of passes through the processing loop. The remainder of the input file is read in lines 5400 to 5475. Note that the noisy velocity values are set to zero as they are read. The output file is opened under the name DVEL or DPOW with the time indicated as in the data files. The date string is also modified and the revised header is written

to the destination disk in lines 5470 to 5555. The remainder of the loop consists of a call to the appropriate calculation routine followed by a write of the output information to disk.

The hourly statistics for power consist of the 50% and 90% power values for all altitudes. These are stored as dB above a reference level which is also in dB. In order to calculate the 50% and 90% values it is necessary to order the data. Furthermore, the minimum for all 20 heights is needed for the reference. A general purpose sort routine based on Floyd's tree is shown in lines 7500 to 7740. An array G of length A1 is to have A2 elements in order after sorting. If A3 is one then these will be the largest elements; a zero for A3 causes a sort for the smallest elements. Finally, the parameter A4 determines if the sorted data should be in the high or low index elements of the G array.

The sort routine is called from the routine in lines 7400 to 7455 which handles the power files. To determine the 50% and 90% power levels only half of the elements, plus one, need to be in order. The number of elements to be sorted for largest value is calculated in lines 7407 and 7408. This operation is not dependent on height but can depend on the number of minutes in the file and therefore appears inside the hour loop from line 7405 to line 7448. In the height loop from line 7415 to line 7441 the required section of the data file is transferred to the G array and sorted. The routine at line 8100 selects the 50% and 90% values form the sorted array. The values are placed in columns 1 and 2 of the E array for the first hour and columns 3 and 4 for the second hour. The row in the array is determined by the height. The remaining unsorted elements of the G array are used to determine the minimum for all heights. The tree-sort routine would require more computation to sort a single element than the straightforward technique

and therefore the subroutine is not used. After all 20 heights have been sorted for a given hour the minimum for that hour is subtracted from the 50% and 90% values. If no data exist for the second hour then the E array and hour minimum are zeroed so that the disk file format is independent of the number of data points. The hourly minima and the E array are output to disk. The format of the DPOW files is illustrated in Table A.22.

The hourly statistics for the velocity data consist of 6 values for each height: the sample mean converted to a horizontal velocity, the standard deviation of the samples, the number of points with measurable velocity, the correlation half-width, the number of independent samples, and the standard deviation of the mean as a horizontal velocity. The velocity calculations occur in lines 7000 to 7235 with the correlation routine in lines 7800 to 7905. The velocity calculations use default values or the value at the altitude above if the number of points at a given altitude and hour is too small or the points are scattered throughout the hour and the correlation is meaningless. The altitude loop therefore proceeds from the highest altitude to the lowest altitude. Only the correlation half-width, the number of independent points, and the standard deviation of the mean can take on default values. Default values can therefore be indicated with a negative sign.

The mean, variance and number of points are calculated in lines 7061 to 7140. This code is written with a minimum of array references which would slow the computation. One of two entry points to the correlation routine is used based on the number of points in the data set. The value returned is the half-width of the correlation function. The number of independent points, calculated in lines 7155 to 7160, equals the number of points multiplied by 0.6643 and divided by the correlation half-width. Finally,

Table A.22 Apple II disk format for DPOW files.

Sequential text file, Apple DOS 3.3
File name e.g. DPOW/1038
Stored on disk with a date file, one day per disk

L$: date and file type, e.g. November 3. 1980 DPOW
H3: start time hours (CST)
M3: start time minutes
N3: number of minutes in the input data file
S2: same as for input data file--see Roth [1982]
L2: same as for input data file
A6: same as for input data file
H2: base altitude; base + 1.5 km = lowest sample altitude

MN(1): minimum power in tenths of a dB for all heights in the first hour of data
MN(2): minimum power in tenths of a dB for the second hour or zero for no data

E(1,1): 50% power value in tenths of a dB above MN(1) for the lowest altitude and the first hour
E(1,2): 90% power value in tenths of a dB above MN(1) for the lowest altitude and the first hour
E(1,3): 50% power value in tenths of a dB above MN(2) for the lowest altitude and the second hour or zero for no data
E(1,4): 90% power value in tenths of a dB above MN(2) for the lowest altitude and the second hour or zero for no data

E(2,1): 50% for second altitude, first hour
E(2,2): 90% for second altitude, first hour
E(2,3): 50% for second altitude, second hour or zero
E(2,4): 90% for second altitude, second hour or zero

....

E(20,1): 50% for highest altitude, first hour
E(20,2): 90% for highest altitude, first hour
E(20,3): 50% for highest altitude, second hour or zero
E(20,4): 90% for highest altitude, second hour or zero

the sample standard deviation, the standard deviation of the mean, and the mean converted to a horizontal velocity are calculated in lines 7165 to 7190. If there is no data for the second hour the second half of the E array (columns 7 to 12) is zeroed. The E array is written to disk. The format of the DVEL files is illustrated in Table A.23.

The programs DPOW READ and DVEL READ, shown respectively in Tables A.24 and A.25, read the DPOW and DVEL files and print them on the Spinwriter. These programs serve only to indicate that the files are stored correctly on disk and are not designed to produce a formatted or labeled output. The DVEL READ program requires 132 column width paper.

A.10 Calculation and Plotting of Measurable Velocity Statistics

The calculation of the percentage of minute-by-minute data points with measurable velocity is based on the DVEL disk files discussed above. In those files the number of possible points, which is the same for all 20 altitudes, and the number of points with velocity values are recorded. The programs illustrated in this section are used to select the appropriate statistics from the DVEL files, store them to disk by altitude, hour of the day, and month, and to print the results in a table.

The NUMPOINTS3 program, shown in Table A.26, reads DVEL files and creates or modifies sum files. Although the calculations are minimal, the program must deal with a large amount of data and therefore has been compiled to NUMPOINTS3.OBJ. Each sum file or POINTS file, illustrated in Table A.27, contains the results for one month. The program requests the month and year of the POINTS file to modify and the slot, drive, and volume of the disk which contains the POINTS file. If the file is not found in line 140 then the routine in lines 2025 to 2090 can be used to create a file for the specified month and year. The slot, drive, and volume for the disk

Table A.23 Apple II disk format for DVEL files.

```
Sequential text file, Apple DOS 3.3
File name e.g. DVEL/1038
Stored on disk with a date file, one day per disk

L$:  date and file type, e.g. November 3. 1980 DVEL
H3:  start time hours (CST)
M3:  start time minutes
N3:  number of minutes in the input data file
S2:  same as for input data file--see Roth [1982]
L2:  same as for input data file
A6:  same as for input data file
H2:  base altitude; base + 1.5 km = lowest sample altitude

E(1,1):    horizontal velocity toward the northwest for the lowest
             altitude and first hour  (m/s)
E(1,2):    standard deviation of the line-of-sight velocity for the
             lowest altitude and first hour  (m/s)
E(1,3):    no. of minutes with measureable velocity for the lowest
             altitude and first hour
E(1,4):    correlation half-width of the line-of-sight velocity
             for the lowest altitude and first hour  (minutes)
E(1,5):    no. of independent samples for the lowest altitude and
             first hour
E(1,6):    standard deviation of the horizontal velocity for the
             lowest altitude and first hour  (m/s)
E(1,7):
E(1,8):
E(1,9):    data in columns 7 to 12 are the same as columns 1 to 6
E(1,10):     respectively, except they are for the lowest altitude
E(1,11):     and the second hour or are zero if no data is available
E(1,12):

E(2,1):
 ....      same column interpretation, second height
E(2,12):

          .....................
          .....................

E(20,1):
 ....      same column interpretation, highest height
E(20,12):
```

Table A.24 DPOW READ.

```
10 D$ = CHR$ (4)
11  INPUT "FILE NAME ? ";F$
15  PRINT D$;"PR#1"
20  PRINT D$;"OPEN ";F$
30  PRINT D$;"READ ";F$
50  FOR I = 1 TO 8
60  INPUT A$
65  PRINT A$
70  NEXT
71  INPUT M1: INPUT M2
72  PRINT M1,M2
75  FOR I = 1 TO 20
77  PRINT I;" ";
80  FOR I2 = 1 TO 4
90  INPUT A$
91  PRINT A$;" ";
110  NEXT
115  PRINT
120  NEXT
121  PRINT : PRINT
150  PRINT D$;"CLOSE"
160  PRINT D$;"PR#0"
```

Table A.25 DVEL READ.

```
10 D$ =  CHR$ (4)
11  INPUT "FILE NAME ? ";F$
15  PRINT D$;"PR#1"
20  PRINT D$;"OPEN ";F$
30  PRINT D$;"READ ";F$
50  FOR I = 1 TO 8
60  INPUT A$
65  PRINT A$
70  NEXT
75  FOR I = 1 TO 20
77  PRINT I;" ";
80  FOR I2 = 1 TO 12
90  INPUT A$
91  PRINT A$;" ";
110  NEXT
115  PRINT
120  NEXT
121  PRINT : PRINT
150  PRINT D$;"CLOSE"
160  PRINT D$;"PR#0"
```

Table A.26 NUMPOINTS3.


```
1  REM  FIND FRACTION OF TIME WITH
2  REM  MEASUREABLE VELOCITIES PER HOUR
3  REM  PER ALTITUDE PER MONTH
10  REM  KEN GIBBS WITH THANKS TO MARY OZARKA
11  REM  7/10/82
20  REM  ! INTEGER TN,NT,NH,IH,HR,I,H3,M3,H4,CM,NF,UF,HN
50 D$ =  CHR$ (4)
55  DIM SA(25,16,2)
57  DIM E(20,12)
60  DIM FT(10),F$(50)
65  REM  ZERO THE NUMBER OF FILES COUNTER
66 NF = 0
100  REM  GET MONTH,YEAR
105  INPUT "MONTH? ";M$
110  INPUT "YEAR? ";YR
115  IF YR < 1978 OR YR > 1989 GOTO 110
120  INPUT "SUM FILE S,D,V ?";DS,DD,DV
125  IF DS < 4 OR DS > 6 OR DD < 0 OR DD > 1 OR DV < 0 GOTO 120
130  ONERR  GOTO 2000
135 DF$ = "POINTS/" + M$ + "." +  STR$ (YR)
140  PRINT D$;"VERIFY ";DF$;",S";DS;",D";DD;",V";DV
150  POKE 216,0
175  PRINT D$;"OPEN ";DF$
180  PRINT D$;"READ ";DF$
185  INPUT M1$
190  INPUT Y1
195  INPUT TS$
200  INPUT HD
205  INPUT NH
220  FOR IH = 1 TO NH
225  FOR HR = 0 TO 16
230  FOR I = 0 TO 2
235  INPUT SA(IH,HR,I)
240  NEXT
245  NEXT
250  NEXT
255  PRINT D$;"CLOSE ";DF$
300  REM  READY TO ADD TO THE ARRAY
305  INPUT "DATA FILE S,D,V ? ";SS,SD,SV
310  IF SS < 4 OR SS > 6 OR SD < 0 OR SD > 1 OR SV < 0 GOTO 305
315  ONERR  GOTO 3000
320  PRINT D$;"CATALOG S";SS;",D";SD;",V";SV
322  POKE 216,0
325  REM  GET TIMES
330 NT = 0
335  INPUT "TIME (E.G. 1029) ? ";TM
340  IF TM = 0 AND NT = 0 THEN  GOTO 9000
345  IF TM = 0 AND NT <  > 0 GOTO 365
350 NT = NT + 1
355 FT(NT) = TM
360  IF NT < 10 GOTO 335
365  REM  GOT THE NT TIMES IN ARRAY FT
499  REM  START READING FILES
500 TN = 0
505  REM  LOOP OVER FILE NAMES BELOW
510 TN = TN + 1
520  REM  FILE NAME FOR INPUT IS SF$
530 SF$ = "DVEL/" +  STR$ (FT(TN))
535  ONERR  GOTO 790
540  PRINT D$;"VERIFY ";SF$
545  POKE 216,0
550  PRINT D$;"OPEN ";SF$
560  PRINT D$;"READ ";SF$
570  INPUT H1$: INPUT H3: INPUT M3
580  INPUT N3: INPUT WC: INPUT WC: INPUT WC
600  INPUT HS
620  FOR IH = 1 TO 20
630  FOR I = 1 TO 12
640  INPUT E(IH,I)
```

Table A.26 cont'd.

```
650  NEXT
660  NEXT
670  PRINT D$;"CLOSE ";SF$
671  REM  CHECK INPUT FOR CORRECT DATE
672  IF  MID$ (M$,3,1) <  >  MID$ (H1$,3,1) OR  MID$ (M$,2,1) <  >  MID$
     (H1$,2,1) THEN  PRINT "CAN'T USE THIS DATA": GOTO 792
673  REM  GET THE YEAR
674 I = 1:SP = 0
675  IF  MID$ (H1$,I,1) <  > " " THEN I = I + 1: GOTO 675
676 SP = SP + 1:I = I + 1
677  IF SP < 2 GOTO 675
678 Y2 =  VAL ( MID$ (H1$,I,4))
679  IF Y2 <  > Y1 THEN  PRINT "CAN'T USE THIS DATA": GOTO 792
680  REM  FIND TIME POINTER IN SA
682 H4 = H3 - 4
685  IF M3 >  = 30 THEN H4 = H4 + 1
690  IF H4 > 16 GOTO 790
695  REM  UF=FLAG TO SHOW IF INPUT FILE USED
700 UF = 0
705  IF N3 <  = 60 GOTO 760
710  IF H4 < 0 GOTO 730
715  REM  FIRST OF TWO HOURS
720 N4 = 60:CM = 3
725  GOSUB 8000
730 H4 = H4 + 1
735  IF H4 < 0 OR H4 > 16 GOTO 780
740  REM  SECOND OF TWO HOURS
745 N4 = N3 - 60:CM = 9
750  COSUB 8000
755  GOTO 780
760  REM  ONE FOUR ONLY
765  IF H4 < 0 OR H4 > 16 GOTO 780
770 N4 = N3:CM = 3
775  GOSUB 8000
780  REM  DONE WITH INPUT FILE
785  IF UF = 1 THEN NF = NF + 1:F$(NF) = H1$ + " " +  STR$ (FT(TN))
790  IF TN < NT GOTO 510
792  INPUT "ADD MORE TO THIS SUMFILE? (Y/N)";Q$
795  IF  LEFT$ (Q$,1) = "N" GOTO 1000
800  IF  LEFT$ (Q$,1) <  > "Y" GOTO 790
805  REM  GET MORE INPUT
810  INPUT "SAME S,D,V FOR INPUT? (Y/N)";Q$
815  IF  LEFT$ (Q$,1) = "N" GOTO 305
820  IF  LEFT$ (Q$,1) = "Y" GOTO 315
825  GOTO 810
1000  REM  WRITE SUMFILE TO DISK
1005  PRINT "INSERT DESTINATION DISK"
1010  INPUT "TYPE C TO CONTINUE...";Q$
1015  IF  LEFT$ (Q$,1) <  > "C" GOTO 1010
1020  ONERR  GOTO 4000
1025  PRINT D$;"OPEN ";DF$;",S";DS;",D";DD;",V";DV
1030  PRINT D$;"DELETE ";DF$
1035  PRINT D$;"OPEN ";DF$
1040  PRINT D$;"WRITE ";DF$
1045  PRINT M1$: PRINT Y1: PRINT TS$: PRINT HD: PRINT NH
1050  FOR IH = 1 TO NH
1055  FOR HR = 0 TO 16
1060  FOR I = 0 TO 2
1065  PRINT SA(IH,HR,I)
1070  NEXT
1075  NEXT
1080  NEXT
1085  PRINT D$;"CLOSE ";DF$
1090  GOTO 9000
2000  REM  CAN'T READ SUMFILE
2002  POKE 216,0
2005  PRINT "CAN'T READ ";DF$
2010  INPUT "CREATE A NEW FILE? (Y/N) ";Q$
2015  IF  LEFT$ (Q$,1) = "N" GOTO 9000
```

Table A.26 cont'd.


```
2020  IF  LEFT$ (Q$,1) <  > "Y" GOTO 2010
2025  REM  SET UP NEW FILE
2030 M1$ = M$
2035 Y1 = YR
2040 TS$ = "POINTSUM"
2045 HD = 57
2050 NH = 25
2055  FOR IH = 1 TO NH
2060  FOR HR = 0 TO 16
2065  FOR I = 0 TO 2
2070 SA(IH,HR,I) = 0
2075  NEXT
2080  NEXT
2085  NEXT
2090  GOTO 300
3000  REM  CAN'T CATALOG DATA DISK
3005  POKE 216,0
3010  PRINT "CAN'T CATALOG AT S,D,V GIVEN"
3015  PRINT "Q)UIT T)RY AGAIN C)HANGE S,D,V"
3020  INPUT "INPUT Q,T, OR C? ";Q$
3025  IF  LEFT$ (Q$,1) = "Q" GOTO 9000
3030  IF  LEFT$ (Q$,1) = "T" GOTO 315
3035  IF  LEFT$ (Q$,1) = "C" GOTO 305
3040  GOTO 3020
4000  REM  CAN'T WRITE SUM FILE OUT
4005  POKE 216,0
4010  PRINT "CAN'T WRITE SUM FILE TO DISK"
4015  PRINT "Q)UIT T)RY AGAIN C)HANGE S,D,V"
4020  INPUT "INPUT Q,T, OR C? ";Q$
4025  IF  LEFT$ (Q$,1) = "Q" GOTO 9000
4030  IF  LEFT$ (Q$,1) = "T" GOTO 1020
4035  IF  LEFT$ (Q$,1) <  > "C" GOTO 4020
4040  INPUT "SUM FILE S,D,V? ";DS,DD,DV
4045  IF DS < 4 OR DS > 6 OR DD < 0 OR DD > 1 OR DV < 0 GOTO 4040
4050  GOTO 1020
8000  REM  ROUTINE TO ADD ONE HOUR'S DATA
8005  REM  CM IS THE COLUMN POINTER IN THE E ARRAY
8010  REM  H4 IS THE HOUR POINTER IN SA ARRAY
8015  REM  N4 IS THE POSSIBLE POINTS FOR THE HOUR
8020  REM  HN IS THE POINTER TO HEIGHT IN SA ARRAY
8025 HN = (HS - HD) / 1.5
8030  FOR IH = 1 TO 20
8035 HN = HN + 1
8040  IF HN > NH GOTO 8070
8045  IF HN < 1 GOTO 8065
8050 SA(HN,H4,1) = SA(HN,H4,1) + N4
8055 SA(HN,H4,0) = SA(HN,H4,0) + E(IH,CM)
8057 SA(HN,H4,2) = SA(HN,H4,2) + 1
8060 UF = 1
8065  NEXT
8070  RETURN
9000  REM  QUIT
9001  IF NF = 0 GOTO 9005
9002  FOR I = 1 TO NF
9003  PRINT F$(I)
9004  NEXT
9005  PRINT "& AND RETURN TO RESTART"
9010  END
```

C - 3

Table A.27 Apple II disk format for NUMPOINTS files.

Sequential text file, Apple DOS 3.3
File name e.g. POINTS/APRIL.1978

M1$: month
Y1: year
TS$: "POINTSUM"--indicates the type of data
HD: 57 = base altitude, first altitude at 58.5 km, etc.
NH: 25 = number of heights

SA(1,0,0): lowest height, 400 CST, number of points with measureable velocity
SA(1,0,1): lowest height, 400 CST, number of possible points
SA(1,0,2): lowest height, 400 CST, number of data files used at this height and hour

SA(1,1,0): lowest height, 500 CST, etc.
SA(1,1,1): lowest height, 500 CST, etc.
SA(1,1,2): lowest height, 500 CST, etc.

....

SA(1,16,0): lowest height, 2000 CST, etc.
SA(1,16,1): lowest height, 2000 CST, etc.
SA(1,16,2): lowest height, 2000 CST, etc.

SA(2,0,0): 2nd height, 400 CST, etc.
SA(2,0,1): 2nd height, 400 CST, etc.
SA(2,0,2): 2nd height, 400 CST, etc.

....

SA(NH,16,0): highest height, 2000 CST, etc.
SA(NH,16,1): highest height, 2000 CST, etc.
SA(NH,16,2): highest height, 2000 CST, etc.

containing the DVEL files and the time of these files are the next input. Recall that the file names are of the form DVEL/1034. The program will accept a maximum of 10 file times although typically there are only 3 times on a data disk. A time which does not correspond to a DVEL file on the disk is ignored as the files are read. The input of file times is terminated by entering a time of 0. There is no provision for showing the file times to the operator while the program is running so it is necessary to have a directory of the data disks available. A list of the files which were used by the program is printed on the screen when the modified POINTS file is written back to disk.

The main loop of the NUMPOINTS3 program reads the DVEL files and sums the values to the appropriate locations in the POINTS file. After reading the DVEL file the date of the data is checked to be certain that it matches the month and year of the POINTS file. The data in the DVEL files are grouped into hours which are mapped into one hour time bins in the POINTS file based on the integral hour value nearest to the start time. The addition of one hour of data is performed by the subroutine in lines 8000 to 8070. Note that there are 25 height bins in the POINTS file to accommodate the variation in the heights which occurs in the DVEL files. Only those heights in the data file which correspond to heights in the POINTS file are used. When a data file has been added to the POINTS file the program loops back to read the next data file until the list of times is exhausted. The operator can choose to write the sum file to disk at that point or to add files from another data disk. In this way the floppy disks for all the days in a given month are added together. After adding all the files of interest the POINTS file is written back to disk using the original slot, drive, and volume parameters. The previous version of the POINTS file is lost.

The structure of the POINTS files consists of a short header followed by a 3-dimensional array stored sequentially with the last index varying most quickly. As shown in Table A.27, for each height and hour there are three elements consisting of the number of points with measurable velocity, the number of possible points, and the number of files which contributed to the data. The number of files indicates the quality of the fraction obtained when the number of actual points is divided by the number of possible points. For each height there are 17 one-hour slots from 400 CST to 2000 CST. The base altitude and number of heights are fixed at 57 km and 25 in the header for all POINTS files used in this work. However, the NUMPOINTS3 program will handle different values for these parameters provided some means to input these two values is included in place of lines 2045 and 2050. The declaration of the SA array should reflect any increase in the number of heights. This allows the program to be used for tropospheric data.

The program PERCENT3.6, shown in Table A.28, reads a POINTS file from disk and prints out the percentage of points with measurable velocities in a tabular form. The number of data files used for each hour and altitude is printed in a separate table. The program uses the escape-code tab function of the Spinwriter to achieve a formatted table in each case. PERCENT3.6 expects 25 heights or less but can accommodate the changes which might be made in NUMPOINTS3 by changing the declaration of the SA array. Although the program is written in a compiler-ready form it is of little advantage to compile a program which requires the greatest percentage of time to print results.

A.11 Generation of the Power Spectra of Minute-by-Minute Data

The program PVCFFTGEN5.1, shown in Table A.29, calculates the power

Table A.28 PERCENT3.6.


```
1  REM  CALCULATE FRACTION OF MEASUREABLE
2  REM  VELOCITIES FROM THE POINTS FILES
10  REM  KEN GIBBS
11  REM  7/6/82
25  REM  ! INTEGER I,IH,HR,NH,NP
50 D$ =  CHR$ (4)
55  DIM SA(25,16,2)
100  REM  GET MONTH,YEAR
105  INPUT "MONTH? ";M$
110  INPUT "YEAR? ";YR
115  IF YR < 1978 OR YR > 1989 GOTO 110
117 DF$ = "POINTS/" + M$ + "." +  STR$ (YR)
120  INPUT "SUM FILE S,D,V ?";DS,DD,DV
125  IF DS < 4 OR DS > 6 OR DD < 1 OR DD > 2 OR DV < 0 GOTO 120
130  ONERR  GOTO 4000
140  PRINT D$;"VERIFY ";DF$;",S";DS;",D";DD;",V";DV
150  POKE 216,0
175  PRINT D$;"OPEN ";DF$
180  PRINT D$;"READ ";DF$
185  INPUT M1$
190  INPUT Y1
195  INPUT TS$
200  INPUT HD
205  INPUT NH
220  FOR IH = 1 TO NH
225  FOR HR = 0 TO 16
230  FOR I = 0 TO 2
235  INPUT SA(IH,HR,I)
240  NEXT
245  NEXT
250  NEXT
255  PRINT D$;"CLOSE ";DF$
260  POKE 216,0
500  REM  CALCUALTE AND PRINT
505  PRINT D$;"PR#1"
510 ES$ =  CHR$ (27)
520 TB$ =  CHR$ (9)
525  PRINT : PRINT
530  PRINT M1$;".";Y1
535  PRINT "PERCENTAGE OF MEASUREABLE VELOCITIES"
540  PRINT
550  GOSUB 2000
700  FOR IH = NH TO 1 STEP  - 1
701 NP = 0
702  FOR HR = 0 TO 16
703  IF SA(IH,HR,2) = 0 GOTO 705
704 NP = 1
705  NEXT
706  IF NP = 0 GOTO 840
710 HT = HD + 1.5 * IH
720  PRINT HT;
750  FOR HR = 0 TO 16
760  IF SA(IH,HR,2) = 0 GOTO 820
770 PC =  INT (100 * SA(IH,HR,0) / SA(IH,HR,1) + 0.5)
780 TP = 7 + HR * 4
790  IF PC < 10 THEN TP = TP + 2
800  IF PC < 100 AND PC > 9 THEN TP = TP + 1
810  PRINT ES$;TB$; CHR$ (TP);PC;
820  NEXT
830  PRINT
840  NEXT
850  PRINT "ALTITUDE"
860  PRINT "  (KM)"
900  PRINT : PRINT
920  PRINT M1$;".";Y1
930  PRINT "NUMBER OF FILES"
940  PRINT
950  GOSUB 2000
960  FOR IH = NH TO 1 STEP  - 1
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.28 cont'd.

```
961 NP = 0
962  FOR HR = 0 TO 16
963  IF SA(IH,HR,2) = 0 GOTO 965
964 NP = 1
965  NEXT
966  IF NP = 0 GOTO 1050
970 HT = HD + 1.5 * IH
980  PRINT HT;
990  FOR HR = 0 TO 16
1000 TP = 8 + HR * 4
1010  IF SA(IH,HR,2) < 10 THEN TP = TP + 1
1020  PRINT ES$;TB$; CHR$ (TP);SA(IH,HR,2);
1030  NEXT
1040  PRINT
1050  NEXT
1060  PRINT "ALTITUDE"
1070  PRINT "  (KM)"
1080  PRINT D$;"PR#0"
1090  GOTO 9000
2000  REM  PRINT HOUR HEADER
2010  PRINT "       TIME(CST)"
2020  FOR HR = 0 TO 16
2030 TP = 8 + HR * 4
2040  IF HR < 6 THEN TP = TP + 1
2050  PRINT ES$;TB$; CHR$ (TP);HR + 4;
2060  NEXT
2070  PRINT
2080  PRINT "       ==========================================================
     ============="
2090  RETURN
4000  REM  CAN'T FIND SUM FILE
4005  POKE 216,0
4010  PRINT "CAN'T FIND SUM FILE"
4015  PRINT "Q)UIT T)RY AGAIN C)HANGE S,D,V"
4020  INPUT "INPUT Q,T, OR C? ";Q$
4025  IF  LEFT$ (Q$,1) = "Q" GOTO 9000
4030  IF  LEFT$ (Q$,1) = "T" GOTO 130
4035  IF  LEFT$ (Q$,1) = "C" GOTO 120
4040  GOTO 4020
9000  REM  QUIT
9005  PRINT "& AND RETURN TO RESTART"
9010  END
```

Table A.29 PVCFFTGEN5.1.


```
1 REM PROGRAM FOR FINDING POWER SPECTRA
2 REM OF CS DATA ON APPLE DISK
3 REM SUBTRACT MEAN AND FILL WITH ZEROES TO 128
4 REM 128 POINT FFT- THEN TAKE MAGNITUDE
5 REM RESULTS FOR EACH HEIGHT ON DISK
7 REM ANTILOG OF POWER FILES IS USED
8 REM SOURCE,DESTINATION MUST BE AT DIFFERENT S,D,V
10 REM KEN GIBBS
11 REM 6/15/82
50 REM ! INTEGER I,J,K,N2,IP,L,L1,LE,LC,N9,N3
55 REM ! INTEGER NT,KL,KH,KN,TN,N,M,N1,FN
60 REM ! INTEGER TM
100 D$ = CHR$ (4)
105 DIM RE(128),IM(128),CR(64),CI(64)
110 N = 128:N2 = N / 2:M = 7:N1 = N - 1
120 DIM FT(10),FK$(3)
122 DIM A(20,120)
131 REM ET IS THE ANTILOG FACTOR FOR POWER
132 ET = LOG (10) / 100
135 REM FILL COEFFECIENTS ONCE ONLY
140 GOSUB 7400
145 REM INIT S,D,V TO ALLOW SOURCE<>DEST TEST
150 SS = 0:SD = 1:SV = 0
155 DS = 0:DD = 1:DV = 0
1000 REM SET RP=4 TO WEAVE THROUGH THE DATE AND S,D,V ROUTINES
1005 RP = 4
1010 GOTO 6600
4000 REM CHECK DISKS
4020 ONERR GOTO 6300
4030 PRINT D$;"VERIFY ";H$;",S"; STR$ (SS);",D"; STR$ (SD);",V"; STR$ (S
     V)
4040 ONERR GOTO 6200
4050 PRINT D$;"VERIFY ";H$;",S"; STR$ (DS);",D"; STR$ (DD);",V"; STR$ (D
     V)
4060 ONERR GOTO 6100
4070 PRINT D$;"OPEN DUMMY"
4080 PRINT D$;"DELETE DUMMY"
5000 REM GET FILE INFO.
5010 REM ZERO FOR TIME INPUT TERMINATES
5020 REM PROGRAM WILL SKIP BAD FILE NAMES
5030 REM SO DON'T WORRY ABOUT ERRORS
5100 NT = 0
5110 INPUT "TIME (E.G. 1029) ? ";TM
5115 IF TM = 0 AND NT = 0 GOTO 5740
5120 IF TM = 0 AND NT < > 0 GOTO 5155
5125 NT = NT + 1
5130 FT(NT) = TM
5135 IF NT < 10 GOTO 5110
5155 KL = 1:KH = 0
5160 INPUT "POWER FILES? Y OR N ";Q$
5170 IF LEFT$ (Q$,1) = "Y" THEN KH = KH + 1:FK$(KH) = "POWW": GOTO 5200

5180 IF LEFT$ (Q$,1) = "N" GOTO 5200
5190 GOTO 5160
5200 INPUT "VELOCITY FILES? Y OR N ";Q$
5210 IF LEFT$ (Q$,1) = "Y" THEN KH = KH + 1:FK$(KH) = "VELL": GOTO 5240

5220 IF LEFT$ (Q$,1) = "N" GOTO 5240
5230 GOTO 5200
5240 INPUT "CORRELATION FILES? Y OR N ";Q$
5250 IF LEFT$ (Q$,1) = "Y" THEN KH = KH + 1:FK$(KH) = "CORR": GOTO 5280

5260 IF LEFT$ (Q$,1) = "N" GOTO 5280
5270 GOTO 5240
5280 IF KL > KH GOTO 5740
5300 TN = 1
5310 KN = KL
5320 F$ = FK$(KN) + "/" + STR$ (FT(TN))
5324 REM SET THE POWER FILE FLAG
```

Table A.29 cont'd.

```
5325  IF  LEFT$ (FK$(KN),1) = "P" THEN PF = 1
5326  IF  LEFT$ (FK$(KN),1) <  > "P" THEN PF = 0
5330  ONERR  GOTO 5660
5340  PRINT D$;"VERIFY ";F$;",S"; STR$ (SS);",D"; STR$ (SD);",V"; STR$ (S
     V)
5345  PRINT "READING ";F$
5350  PRINT D$;"OPEN ";F$
5360  PRINT D$;"READ ";F$
5370  INPUT L$: INPUT H3: INPUT M3: INPUT N3
5380  INPUT S2: INPUT L2: INPUT A6: INPUT H2
5400  FOR I = 1 TO 20
5410  FOR J = 1 TO 120
5420  INPUT Z
5424  REM  TAKE ALOG OF POWER FILES
5425  IF PF = 1 THEN A(I,J) =  EXP (ET * Z): GOTO 5440
5430  IF Z > 8000 THEN A(I,J) = 0: GOTO 5440
5435 A(I,J) = Z
5440  NEXT
5450  NEXT
5460  PRINT D$;"CLOSE ";F$
5470  REM  DONE WITH INPUT, START OUTPUT WITH HEADER
5475  PRINT "DONE READING ";F$
5480 F1$ = "FFT" +  LEFT$ (FK$(KN),1) + "/" +  STR$ (FT(TN))
5485  PRINT F1$
5490  ONERR  GOTO 6030
5500  PRINT D$;"OPEN ";F1$;",S"; STR$ (DS);",D"; STR$ (DD);",V"; STR$ (DV
     )
5510  PRINT D$;"DELETE ";F1$
5515  ONERR  GOTO 6000
5520  PRINT D$;"OPEN ";F1$
5530  PRINT D$;"WRITE ";F1$
5535  REM  CHANGE L$
5540 L$ = H$ + " " +  LEFT$ (F1$,4)
5545  PRINT L$: PRINT H3: PRINT M3: PRINT N3
5550  PRINT S2: PRINT L2: PRINT A6: PRINT H2
5570  FOR IH = 1 TO 20
5580  GOSUB 7300: REM  FIND AND SUBTRACT MEAN, FILL G
5599  REM  SKIP FFT IF ALL ZERO
5600  IF PN > 1 THEN  GOSUB 7500: REM  FFT
5606  GOSUB 8300: REM  MAGNITUDE SQUARED,NORMALIZE
5618  PRINT PN: REM  THE NO. OF POINTS
5619  PRINT FU: REM  THE FUDGE FACTOR
5620  FOR I = 1 TO 65
5630  PRINT RE(I)
5640  NEXT
5650  NEXT
5660  PRINT D$;"CLOSE ": REM  CLOSES INPUT FILE ONERR, OUTPUT NORMALLY
5670  POKE 216,0
5680 KN = KN + 1
5690  IF KN <  = KH GOTO 5320: REM  SAME TIME,NEW KIND
5700  REM  DONE ALL FILES WITH THIS TIME
5710 TN = TN + 1
5720  IF TN <  = NT GOTO 5310: REM  NEW TIME
5730  REM  DONE ALL FILES
5740  PRINT "& AND RETURN TO RESTART"
5750  END
6000  REM  DESTINATION DISK FULL
6010  REM  S,D,V SET TO DESTINATION
6015  PRINT D$;"CLOSE ";F1$
6020  PRINT D$;"DELETE ";F1$
6030  REM  THIS ENTRY FOR UNABLE TO OPEN
6040  PRINT "NO ROOM FOR FILE ";F1$
6050  PRINT "FIX A NEW DISK AND THEN RUN"
6060  PRINT "THIS PROGRAM AGAIN WITH THE"
6070  PRINT "WITH THE REVISED LIST OF FILES"
6080  GOTO 5740
6100  REM  DESTINATION IS WRITE PROTECTED
6110  PRINT "CAN'T WRITE TO DESTINATION"
6120 RP = 1: REM  RETURN POINTER TO 4060
```

Table A.29 cont'd.


```
6130  GOTO 6800
6200  REM  WRONG DESTINATION
6210  PRINT H$;" IS NOT AT DESTINATION S,D,V"
6220 RP = 2: REM  RETURN POINTER TO 4040
6230  GOTO 6800
6300  REM  WRONG SOURCE
6310  PRINT H$;" IS NOT AT SOURCE S,D,V"
6320 RP = 3: REM  RETURN POINTER TO 4020
6330  GOTO 6800
6400  REM  SOURCE S,D,V INPUT
6410  INPUT "SOURCE S,D,V ? ";SS,SD,SV
6420  IF SS > 6 OR SS < 4 OR SD < 1 OR SD > 2 OR SV < 0 GOTO 6410
6425  IF SS = DS AND SD = DD AND SV = DV THEN  GOSUB 8000: GOTO 6410
6430  IF RP = 4 GOTO 6500
6435 RP = 3: REM  FORCE CHECKS
6440  GOTO 6800
6500  REM  DESTINATION S,D,V INPUT
6510  INPUT "DESTINATION S,D,V ? ";DS,DD,DV
6520  IF DS < 1 OR DS > 6 OR DD < 1 OR DD > 2 OR DV < 0 THEN 6500
6525  IF SS = DS AND SD = DD AND SV = DV THEN  GOSUB 8000: GOTO 6510
6530  IF RP = 4 GOTO 4000
6540 RP = 2: REM  FORCE DEST. CHECK
6550  GOTO 6800
6600  REM  GET DATE
6610  INPUT "DATE (E.G. MAY 14. 1980)? ";H$
6620  IF RP = 4 GOTO 6400
6625 RP = 3: REM  FORCE CHECKS FOR DATE CHANGE
6630  GOTO 6800
6800  REM  HANDLE DISK SET UP ERRORS
6810  PRINT "DO YOU WANT TO: "
6820  PRINT "  1. CHANGE DATE"
6830  PRINT "  2. CHANGE SOURCE S,D,V"
6840  PRINT "  3. CHANGE DEST. S,D,V"
6850  PRINT "  4. NO CHANGE, TRY AGAIN"
6860  PRINT "  5. QUIT"
6870  INPUT "INPUT 1 TO 5 ";Q
6880  IF Q < 1 OR Q > 5 THEN  PRINT : GOTO 6800
6890  ON Q GOTO 6600,6400,6500,6900,5740
6900  REM  NOW REENTER MAIN FLOW
6910  ON RP GOTO 4060,4040,4020
6920  PRINT : PRINT "ERROR IN RP CONTROL"
6930  STOP
7300  REM  FIND MEAN,SUBTRACT,FILL RE,IM
7310 AV = 0
7315 PN = 0
7320  FOR I = 1 TO N3
7325  IF A(IH,I) = 0 GOTO 7340
7330 AV = AV + A(IH,I)
7335 PN = PN + 1
7340  NEXT
7345  IF PN <  = 1 GOTO 7360
7350 AV = AV / PN
7360  FOR I = 1 TO N3
7361  REM  ZERO IM IN ANY CASE
7362 IM(I) = 0
7364  REM  IGNORE POINTS WITH NO DATA
7365  IF A(IH,I) = 0 THEN RE(I) = 0: GOTO 7380
7369  REM  OTHERWISE SUBTRACT AVERAGE
7370 RE(I) = A(IH,I) - AV
7380  NEXT
7381  REM  ZERO THE REST OF ARRAY
7382  FOR I = N3 + 1 TO N
7383 RE(I) = 0
7384 IM(I) = 0
7385  NEXT
7390  RETURN
7400  REM  FILL COEFFECIENT ARRAYS
7410 PI = 3.14159265
7420 F1 = PI / N2
```

Table A.29 cont'd.


```
7430 ANGLE = 0
7440  FOR I = 1 TO N2
7450 CR(I) =  COS (ANGLE)
7460 CI(I) =  -  SIN (ANGLE)
7470 ANGLE = ANGLE + F1
7480  NEXT
7490  RETURN
7500  REM  FFT FOR REAL ONLY, POWER OF 2, IN PLACE
7510  REM  SPECIAL CODE STAGES 1 AND 2
7520  REM  COEFFECIENTS CALCULATED EXTERNALLY
7530  REM  RE, IM ARRAYS, N=NO. OF POINTS
7540  REM  N2=N/2,N=2**M,N1=N-1
7550  REM  SHUFFLE
7560 J = 1
7570  FOR I = 1 TO N1
7580  IF I > = J GOTO 7620
7590 TR = RE(J)
7600 RE(J) = RE(I)
7610 RE(I) = TR
7620 K = N2
7630  IF K > = J GOTO 7670
7640 J = J - K
7650 K = K / 2
7660  GOTO 7630
7670 J = J + K
7680  NEXT
7690  REM  END SHUFFLE, START STAGE 1
7700  FOR I = 1 TO N STEP 2
7710 IP = I + 1
7720 TR = RE(IP)
7730 RE(IP) = RE(I) - TR
7740 RE(I) = RE(I) + TR
7750  NEXT
7760  REM  STAGED 2, TWO PARTS
7770  FOR I = 1 TO N STEP 4
7780 IP = I + 2
7790 TR = RE(IP)
7800 RE(IP) = RE(I) - TR
7810 RE(I) = RE(I) + TR
7820  NEXT
7830  FOR I = 2 TO N STEP 4
7840 IP = I + 2
7850 TI =  - RE(IP)
7860 RE(IP) = RE(I)
7870 IM(IP) =  - TI
7880 IM(I) = TI
7890  NEXT
7900  REM  STAGES 3 TO M
7905 LE = 4
7910  FOR L = 3 TO M
7915 L1 = LE
7920 LE = LE + LE
7925 N9 = N2 / L1
7930 LC = 1
7935  FOR J = 1 TO L1
7940 UR = CR(LC)
7945 UI = CI(LC)
7950 LC = LC + N9
7955  FOR I = J TO N STEP LE
7960 IP = I + L1
7965 TR = RE(IP) * UR - IM(IP) * UI
7970 TI = RE(IP) * UI + IM(IP) * UR
7975 RE(IP) = RE(I) - TR:IM(IP) = IM(I) - TI
7980 RE(I) = RE(I) + TR:IM(I) = IM(I) + TI
7985  NEXT : NEXT : NEXT
7990  RETURN
8000  REM  SOURCE AND DEST. DIFFERENT
8005  REM  TO MAKE LIFE SIMPLE
```

Table A.29 cont'd.

```
8010  PRINT "CAN'T HAVE SAME S,D,V"
8015  PRINT "FOR SOURCE AND DEST."
8020  RETURN
8100  REM  PRINT OUT RE,IM ARRAYS
8102  REM  TEST ROUTINE
8105  FOR I = 1 TO 128
8110  PRINT I;"  ";RE(I);"  ";IM(I)
8115  NEXT
8120  RETURN
8300  REM  SQUARE,NORMALIZE
8301  REM  FIRST HALF ONLY SINCE EVEN
8302  IF FN < 2 THEN FU = 1: GOTO 8345
8304 MX = 0
8305  FOR I = 1 TO 65
8307 RE(I) = RE(I) * RE(I) + IM(I) * IM(I)
8310  IF RE(I) > MX THEN MX = RE(I)
8315  NEXT
8317  IF MX = 0 THEN FU = 1: GOTO 8345
8320 FU = 999 / MX
8325  FOR I = 1 TO 65
8330 RE(I) =  INT (FU * RE(I))
8340  NEXT
8345  RETURN
```

spectra for the power, velocity, and correlation minute-by-minute data. For each of the 20 altitudes the mean is subtracted and zeroes are added to the end of the time series to obtain a length of 128. A power of two FFT algorithm is used to calculate the transform. The transform is then squared to obtain the power spectrum. The power data are converted from dB back to a linear form before calculating the spectrum. Noisy velocity values are set to zero.

PVCFFTGEN5.1 handles the input and output disks in the same manner as the program DGEN3 shown above. The routines in lines 6000 to 6930 which handle the disk parameters and the date are therefore identical. The flag RP is used to control the flow through the various input request and error handling routines. The input of file times and file types to be processed occurs in lines 5000 to 5280. There are two major loops in the program, both of which are not FOR-NEXT loops to allow reentry after errors. The file-type loop runs from line 5320 to line 5700 and the file-time loop runs from line 5310 to line 5730. Inside both loops are the statements to read the input file (lines 5340 to 5475) and the subroutine calls and statements to calculate the FFT, square it, and write to the output disk.

The FFT algorithm shown in lines 7500 to 7990 is a modification of the routine used for the RANDOMFFT8 program shown above. In this case the input data are real rather than complex. This allows the special code for the first and second stages of the transform to be further modified in order to reduce the number of calculations. Speed considerations are important for a program which must calculate 20 transforms of length 128 for each input file. The program has therefore been compiled to PVCFFTGEN5.1OBJ. Other routines specific to this program include the subtraction of the mean in lines 7300 to 7390, the calculation of the transform coefficients in lines

7400 to 7490, and the squaring, normalization and output routines.

The files generated by this program are called FFTP, FFTV, or FFTC files depending on the input data. Other than the exponentiation of the power data to return it to a linear form all three types of data are processed identically. The structure of the files is also the same as illustrated in Table A.30. Because the input data are real, the power spectrum will be even. Therefore, only the frequencies up to the folding frequency are saved on disk. The normalization of the files is identical to that used for the spectra of coherently integrated data.

A.12 Programs to Print and Plot the FFTP, FFTV, and FFTC Files

The FFTPVCREAD program, shown in Table A.31, prints out a FFTP, FFTV, or FFTC file on the printer. This program is used to verify the correct storage of a file. The printout is extremely lengthy and therefore the program is used only when a modification is made to one of the storage programs. With the program as shown the output will consist of the data from all 20 spectra, one for each height. This can be reduced by modifying line 75. The program does not accept slot, drive, or volume parameters. In order to point to the correct disk the date file on the floppy disk containing the spectra should be verified before starting the program.

The FFTPVCPLOT2.SRC program, shown in Table A.32, reads a FFTP, FFTV, or FFTC file from disk and transfers the file or a section of the file to the HP-9830 for plotting. Typically the operator uses the catalog option of the program to print a catalog of the floppy disk and then enters the desired file name. The header is read from the file and then the file is closed so that keyboard input can be used to indicate the desired portion of the file. The spectra at altitudes above and including the specified height number are transferred to the HP-9830. The header is read first in

Table A.30 Apple II disk format for FFTP, FFTV and FFTC files.

Sequential text file, Apple DOS 3.3
File name e.g. FFTV/1038
Stored on disk with a date file, one day per disk

L$: date and file type, e.g. November 3. 1980 FFTV
H3: start time hours (CST)
M3: start time minutes
N3: number of minutes in the input data file
S2: same as for input data file--see Roth [1982]
L2: same as for input data file
A6: same as for input data file
H2: base altitude; base + 1.5 km = lowest sample altitude

PN(1): no. of valid data points used to find the FFT at the lowest altitude
FU(1): scaling factor for the spectrum at the lowest altitude
RE(1,1): DC component of spectrum at lowest altitude
....
RE(1,65): component of spectrum at the folding frequency for the lowest altitude

PN(2): no. of valid data points for 2nd altitude
FU(2): scaling factor for 2nd altitude
RE(2,1): DC component of spectrum for 2nd altitude
....
RE(2,65): folding frequency component for 2nd altitude

....
....

PN(20): no. of valid data points for highest altitude
FU(20): scaling factor for highest altitude
RE(20,1): DC component for highest altitude
....
RE(20,65): folding frequency component for highest altitude

Table A.31 FFTPVCREAD.

```
10 D$ =  CHR$ (4)
11  INPUT "FILE NAME ? ";F$
15  PRINT D$;"PR#1"
20  PRINT D$;"OPEN ";F$
30  PRINT D$;"READ ";F$
50  FOR I = 1 TO 8
60  INPUT A$
65  PRINT A$
70  NEXT
75  FOR I = 1 TO 20
76  INPUT IH: INPUT FU
77  PRINT I,IH,FU
80  FOR I2 = 1 TO 65
90  INPUT A$
95  IF I2 = 1 THEN  PRINT "DC",A$: GOTO 110
100  PRINT 128 / (I2 - 1),A$
110  NEXT
120  NEXT
121  PRINT : PRINT
150  PRINT D$;"CLOSE"
160  PRINT D$;"PR#0"
```

Table A.32 FFTPLOT2.SRC.

```
1  REM  READS THE FFT(P,V, OR C) FILES
2  REM  AND TRANSFERS THEM TO HP FOR PLOT
3  REM  RUDY REAVIS
4  REM  07:26:82:16:15
15 D$ =  CHR$ (4): REM  CTRL-D FOR DOS
20  DIM RE(65),S$(50),L$(50),T$(10)
30 VD = 5 / 6 * 100: REM  % THRESHOLD
50  REM  GET FILE NAME & OPEN FILE
60  PRINT : PRINT : PRINT "C)ATALOG F)ILE INPUT Q)UIT"
65 K = 0
70  INPUT "C,F OR Q?";Q$
80  IF  LEFT$ (Q$,1) = "Q" GOTO 1100
90  IF  LEFT$ (Q$,1) = "F" GOTO 120
100  IF  LEFT$ (Q$,1) = "C" GOTO 1200
110  GOTO 60
120  INPUT "FILE NAME?";F$
130  ONERR  GOTO 1000
140  PRINT D$,"VERIFY";F$: POKE 216,0
150  PRINT D$;"OPEN";F$: PRINT D$;"READ";F$
160  REM
161  REM  GET HEADER DATA
162  INPUT L$: REM  TITLE
164  INPUT H3: REM  HOURS
166  INPUT M3: REM  MINUTES
168  INPUT N3: REM  # OF RECORDS
170  INPUT S2
172  INPUT L2
174  INPUT A6
176  INPUT H2: REM  BASE ALTITUDE
179  PRINT D$;"CLOSE";F$
180  GOSUB 2500
192  GOSUB 2000: REM  SET-UP VIA (6522)
195  GOSUB 500: REM  SEND HEADER TO HP
197  REM
198  REM  READ DATA SUBROUTINE
199  REM
200  FOR HT = 1 TO 20: REM 315 NEXT
205  INPUT PN: REM  #OF POINTS
210  INPUT FU: REM  FUDGE FACTOR
215  FOR I1 = 1 TO 65: INPUT RE(I1): NEXT
220  IF PN = 0 THEN  GOTO 315
225 PC = (PN / N3) * 100: IF PC < VD THEN  GOTO 315
226  IF HT < K THEN 315
230  REM
231  REM  SEND THIS DATA @ HEIGHT HT
232  REM
238  REM  SEND DATA CONSTANTS
239  REM  TELL HP THAT DATA IS COMING
240 S$ = "1": GOSUB 800
244  REM  SEND HEIGHT
245 S$ =  STR$ (H2 + 1.5 * HT): GOSUB 800
250 S$ =  STR$ (PN): GOSUB 800
255 S$ =  STR$ (FU): GOSUB 800
259  REM  SEND DATA POINTS
260  FOR I1 = 1 TO 65
265 S$ =  STR$ (RE(I1)): GOSUB 800
270  NEXT
315  NEXT
316 S$ = "0": GOSUB 800: REM  HALT HP
317  PRINT D$;"CLOSE";F$
320  REM  DONE-RESTART
325  RUN
330  STOP
500  REM  SEND HEADER ROUTINE
505  PRINT : PRINT "START HP NOW"
510 S =  LEN (L$) - 5:S$ =  LEFT$ (L$,S): GOSUB 800
515 S$ =  RIGHT$ (L$,4): GOSUB 800
520 T$ =  STR$ (100 * H3 + M3 + 10000)
525 S$ =  RIGHT$ (T$,4): GOSUB 800
```

Table A.32 cont'd.

```
530 S$ =  STR$ (N3): GOSUB 800
555  RETURN
797  REM
798  REM  SEND S$ TO HP SUBROUTINE
799  REM
800 A1 =  LEN (S$)
805  FOR A2 = 1 TO A1
810  IF  PEEK ( - 15475) <  > 19 THEN 810
820  POKE  - 15488, ASC ( MID$ (S$,A2,1)): NEXT
830  IF  PEEK ( - 15475) <  > 18 THEN 830
840  POKE  - 15488,10
850  REM  -15488 IS B DATA REGISTER
860  RETURN
1000  PRINT "NO SUCH FILE ON DISK": GOTO 60
1100  PRINT : PRINT "& AND RETURN TO RESTART": END
1200  ONERR  GOTO 1220
1210  PRINT D$;"CATALOG": POKE 216,0: GOTO 60
1220  PRINT "ERROR TRYING TO CATALOG": POKE 216,0: GOTO 60
2000  REM  INIT VIA PORT B OF 6522
2005  POKE  - 15486,255: REM   DDRB=$FF
2010  POKE  - 15476,136: REM   PCR=$88
2015  POKE  - 15475,255: REM   CLEAR IFR
2020  POKE  - 15474,127: REM   CLEAR IER
2050  RETURN
2500  PRINT "BASE=";H2
2510  INPUT "DO ALL HEIGHTS (Y OR N)?";H$
2520  IF H$ = "Y" THEN 2550
2525  IF H$ <  > "N" THEN 2510
2530  PRINT "ALT=BASE+(1.5*K) K=1,2,3,...20"
2535  INPUT "START AT HEIGHT(INPUT K)";K
2540  IF K < 0 OR K > 20 THEN 2530
2550  PRINT D$;"OPEN";F$
2555  PRINT D$;"READ";F$
2562  INPUT L$: REM   TITLE
2564  FOR I = 1 TO 7
2566  INPUT H
2568  NEXT
2999  RETURN
```

order to associate height numbers in the file with the sample altitudes. The interface is set-up in lines 2000 to 2050 and the header sent to the HP-9830 in lines 500 to 555. The main loop, lines 200 to 315, reads one spectrum from the disk at a time. The spectrum is sent to the HP-9830 if the loop index equals or exceeds the minimum height of interest. To handle the varying number of data points sent to the HP-9830 a software handshake is used. A one is sent to the HP-9830 as the first character of each spectrum. A zero is sent to indicate the end of data. The routine in lines 800 to 860 sends a string S$ to the HP-9830 by determining its length, sending a character at a time, and then sending an ASCII 10 to indicate end of record. A number is sent by converting it to a string and sending it as S$ to the subroutine. Therefore, both strings and numbers can be sent using a single transfer routine. The corresponding HP-9830 program is shown below.

The HP-9830 FFTPVC PLOT program, shown in Table A.33, is a modification of the FFTA PLOT program shown above for the spectra of coherently integrated data. It has been changed to print only the first half of the spectrum and to skip over the statements (lines 1000 to 1110) which would otherwise label the horizontal axis. The modifications to correctly label the horizontal axis would indicate the period in minutes corresponding to a given frequency. Minor changes have also been made in the plot labels.

A.13 Programs to Compare Power and Velocity Variations

The program CSCORRSRC3, shown in Table A.34, computes the cross-correlation of arbitrary sections of the minute-by-minute data files. This includes autocorrelation of a file section or the cross-correlation of different types of files, i.e., power and velocity data. The program is used to study the relationship between variations in power and velocity for

Table A.33 HP-9830 FFTPVC plot program.

```
10 REM PROGRAM TO PLOT FFTPVC FILES
20 REM KEN GIBES 7/21 82
21 REM WITH THANKS TO RUDY
30 REM PROGRAM SHOULD BE RUN WIHEN INDICATED BY APPLE
40 DIM N[3],S[65]
50 DIM M$[25],T$[10],Y$[4]
55 DIM H$[10],L$[40]
56 DIM G$[25],N$[10]
60 DIM P[34]
100 REM READ HEADER
110 ENTER (2,*)G$
115 REM DATE AS STRING
120 ENTER (2,*)N$
125 REM FFT DATA TYPE
130 ENTER (2,*)T$
131 T$[LEN(T$)+1]=" CST"
135 REM TIME AS STRING
140 ENTER (2,*)N3
145 REM NO. OF POSSIBLE POINTS
200 REM FIX DATE
205 L1=POS(G$,",")
210 M$=G$[1,L1-1]
215 M$[L1]=","
220 M$[L1+1]=G$[L1+1]
230 L$="PLOT OF DATA FILE"
300 REM DEFINE PLOT CONTROL
301 REM P(1),P(2),P(3),P(4) ARE SCALE CONTROL
302 REM PAPER IS 8.5 INCH VERTICAL, 11 INCH WIDE
303 REM P(21)=PAPER HEIGHT/WIDTH
304 REM P(5),P(6),P(7)=BOTTOM,TOP,NO. OF TICS ON VERT. AXIS
305 REM P(8)=HORIZ. POSITION OF VERT. AXIS
306 REM P(9)=SPACING BETWEEN HORIZ. TICS
307 REM P(24),P(18)=LEFT, WIDTH OF HORIZ. AXIS
308 REM P(10),P(11)=H,V COORD. FOR DATE
309 REM P(12),P(13)=H,V FOR TIME
310 REM P(14),P(15)=H,V FOR ALTITUDE
311 REM P(16),P(17)=H,V FOR TITLE
312 REM P(19),P(20)=HEIGHT,ASPECT RATIO OF LABELS
313 REM P(22),P(23)=HORIZ. HALF LENGTH,SPACING OF VERT. TICS
314 REM P(25)=HORIZ. SPACE BETWEEN DATA POINTS
315 REM P(26)=VERT.SCALE FOR DATA
316 REM P(27)=PLOT CONTROL 1=DOT,0=LINES
317 REM P(28)=RELATIVE VERT. OF HORIZ. AXIS LABEL TO AXIS
318 REM P(30),P(31)=H,V FOR PERCENT
319 REM P(29)=SIZE OF AXIS LABEL
320 REM P(32)=V FOR FREQ. AXIS
321 REM P(33),P(34)=H,V FOR FUDGE FACTOR
345 REM READ TABLE, ZERO INTO P(I) WHICH ARE CALCULATED
350 FOR I=1 TO 34
355 READ P[I]
360 NEXT I
370 DATA 0,1100,35,850,200,750,10,0,0,75,775,75,700,75,625
372 DATA 50,40,700,2,1.7,0,10,0,300,0,0,0,-25,1,60
374 DATA 550,150,75,475
400 REM CALCULATE OTHER PLOT PARAMETERS
404 REM DUMMY TIC FOR NOW
405 P[9]=4*P[18]/(65-1)
409 REM PUT VERT AXIS AT LEFT OF HORIZ. AXIS
410 P[8]=P[24]
414 REM FIND VERT TIC SPACING
415 P[23]=(P[6]-P[5])/P[7]
419 REM PAPER RATIO USE SCALE STUFF
420 P[21]=(P[4]-P[3])/(P[2]-P[1])
424 REM CALCULATE HORIZ SPACING FOR DATA
425 P[25]=P[18]/(65-1)
429 REM CALCUALTE VERT. MULTIPLIER FOR DATA
430 P[26]=(P[6]-P[5])/999
450 REM FIRST CHARACTER SENT IS (J)UMP
451 REM J1=0 MEANS NO MORE DATA
452 REM J1=1 GET READY FOR DATA
455 ENTER (2,*)J1
457 IF J1=0 THEN 1500
```

Table A.33 cont'd.

```
460 REM READ HEIGHT
465 ENTER (2,*)H$
470 REM READ DATA CONSTANTS
472 ENTER (2,*)N1
473 REM NO. OF ACTUAL POINTS
474 ENTER (2,*)N2
475 REM FUDGE FACTOR
480 REM READ DATA POINTS
482 FOR I=1 TO 65
484 ENTER (2,*)S[I]
485 NEXT I
490 REM FIX HEIGHT STRING
492 H$[LEN(H$)+1]=" M"
500 REM START PLOT
505 SCALE P[1],P[2],P[3],P[4]
510 YAXIS P[5],P[9],P[24],P[24]+P[18]
524 REM TIC VERT AXIS
525 FOR I=1 TO P[7]
527 YAXIS P[5]+I*P[23],+1,P[8],P[8]-P[22]
528 YAXIS P[5]+I*P[23],-1,P[8],P[8]+P[22]
529 NEXT I
530 PLOT P[10],P[11],1
531 LABEL (*,P[19],P[20],0,P[21])M$
535 PLOT P[12],P[13],1
536 LABEL (*)T$
537 FORMAT F4.0," %"
540 PLOT P[14],P[15],1
541 LABEL (*)H$
542 PLOT P[30],P[31],1
543 LABEL (587)100*N1/N3
544 PLOT P[33],P[34],1
545 LABEL (*)N2
547 PLOT P[16],P[17],1
548 LABEL (*)L$[1,8]&N$&L$[3,17]
600 REM PLOT DATA
610 FOR I=2 TO 65
655 REM FIND HORIZONTAL POSITION
660 X=P[24]+(I-1)*P[25]
665 REM FIND VERTICAL POSITION
670 Y=S[I]*P[26]+P[5]
675 REM PUT PEN DOWN AFTER MOVING
680 PLOT X,Y,-2
685 REM CHECK PLOT CONTROL FOR DOTS
690 IF P[27]=0 THEN 700
695 PEN
700 NEXT I
702 PEN
705 GOTO 1130
1000 REM LABEL HORIZ. AXIS
1001 LABEL (*,P[29],P[10],0,P[21])
1005 Y=P[5]+P[28]
1006 PLOT P[24]+P[18],Y,1
1007 CPLOT 1.5,0
1008 LABEL (*)"M/S"
1010 FOR I=2 TO -2 STEP -1
1015 I1=5*I
1020 PLOT P[24]+I1*P[9],Y,1
1025 CPLOT P[I+3],0
1030 LABEL (1035)I1
1035 FORMAT F3.0
1040 NEXT I
1060 REM FREQ. AXIS
1065 XAXIS P[32],P[18]/4,P[24]-P[18],P[24]+P[18]
1066 Y=P[32]+P[28]
1067 PLOT P[24]+P[18],Y,1
1068 CPLOT 1.5,0
1069 LABEL (*)"HZ"
1070 FOR I=4 TO -4 STEP -1
1075 PLOT P[24]+I*P[18]/4,Y,1
1080 IF I >= 0 THEN 1095
1085 CPLOT -1,0
1090 GOTO 1100
1095 CPLOT -1.3,0
1100 LABEL (1105)I
1105 FORMAT F2.0
1110 NEXT I
1130 REM READY FOR MORE?
1135 REM SHOULD LOOP BACK HERE AFTER OK
1140 DISP "PREPARE PLOTTER--TYPE CONT EXEC"
1145 STOP
1150 GOTO 450
1500 END
```

Table A.34 CSCORRSRC3.


```
1 REM  CS DATA CORRELATOR-KEN GIBBS 3/31/82
2 REM  ! INTEGER RM,LM,XS,HT
3 REM  ! INTEGER H3,M3,TO,H4,M4,T1,H5,M5,T2,I1,NV(2),NM
4 REM  ! INTEGER I,J,LL,UL,NL,LG,M(2),AU,PS
5 DIM CD$(2),WN$(2),F$(2),H$(2),T1$(2),T2$(2),HG$(2),A$(20,120)
6 DIM SD(2),R(120),R1(120),G(2,120)
8 D$ =  CHR$ (4):PS = 1:AU = 0
9 ES$ =  CHR$ (27):TB$ =  CHR$ (9)
10  HOME : PRINT "CS DATA CORRELATOR  3/31/82": GOSUB 8100
11  GOSUB 8600: REM  S,D,V
15  GOSUB 8650: REM  DATE
16  ONERR  GOTO 8200
17  PRINT D$;"VERIFY ";H$(PS);",S"; STR$ (SL);",D"; STR$ (DR);",V"; STR$
    (VL)
20  REM  HAVE CORRECT FLOPPY
21  HOME : INPUT "C)ATALOG  F)ILE INPUT  Q)UIT  ";Q$
22  IF  LEFT$ (Q$,1) = "Q" GOTO 8000
23  IF  LEFT$ (Q$,1) = "F" GOTO 30
24  IF  LEFT$ (Q$,1) = "C" GOTO 6000
25  GOTO 21
30  HOME : INPUT "INPUT FILE NAME  ";F$(PS)
32  HOME : PRINT "READING FILE ";F$(PS)
34  ONERR  GOTO 8300
45  PRINT D$;"OPEN ";F$(PS): PRINT D$;"READ ";F$(PS)
46  REM  GET THE HEADER STUFF
47  INPUT L$: INPUT H3: INPUT M3: INPUT N3
49  INPUT S2: INPUT L2: INPUT A6
50  INPUT H2
51  REM  GET DATA ARRAY AS STRINGS
55  FOR I = 1 TO 20
60  FOR J = 1 TO 120
65  INPUT A$(I,J)
66  REM  REMOVE 9000 BIAS LATER
70  NEXT
75  NEXT
80  PRINT D$;"CLOSE ";F$(PS)
100  REM  GET TIME,ALTITUDE INFO
110  REM  TO SUCH THAT T-TO=INDEX FOR THAT T
120 TO = H3 * 60 + M3 - 1
121  IF PS = 1 GOTO 130
122  HOME : INPUT "SAME START TIME? Y OR N ";Q$
123  IF  LEFT$ (Q$,1) = "Y" GOTO 170
124  IF  LEFT$ (Q$,1) = "N" GOTO 130
125  GOTO 122
130  HOME : INPUT "START TIME? (HOURS,MINUTES) ";H4,M4
140  IF H4 > 24 OR H4 < 0 OR M3 < 0 OR M3 > 59 GOTO 130
150  REM  WE WILL NEED TO CLEAN UP THE SCREEN FOR ERROR CONDITIONS
160  REM  SET TEXT WINDOW AFTER DOING A CATALOG AND THEN HOME
170 T1 = H4 * 60 + M4
180  IF T1 <  = TO GOTO 130: REM  TOO EARLY
190  IF T1 > TO + N3 GOTO 130: REM  TOO LATE
200  REM  START TIME OK
201  IF PS = 1 GOTO 210
202  HOME : INPUT "SAME END TIME? Y OR N ";Q$
203  IF  LEFT$ (Q$,1) = "Y" GOTO 220
204  IF  LEFT$ (Q$,1) = "N" GOTO 210
205  GOTO 202
210  HOME : INPUT "END TIME? (HOURS,MINUTES) ";H5,M5
215  IF H5 > 24 OR H5 < 0 OR M5 < 0 OR M5 > 59 GOTO 210
220 T2 = H5 * 60 + M5
230  IF T2 < T1 GOTO 210: REM  TOO EARLY
240  IF T2 > TO + N3 GOTO 210: REM  TOO LATE
244  REM  SAVE TIMES
245 T1$(PS) =  STR$ (100 * H4 + M4):T2$(PS) =  STR$ (100 * H5 + M5)
250  REM  END TIME OK, GET ALTITUDE
251  IF PS = 1 GOTO 260
252  HOME : INPUT "SAME ALTITUDE? Y OR N ";Q$
253  IF  LEFT$ (Q$,1) = "N" GOTO 260
254  IF  LEFT$ (Q$,1) = "Y" THEN A1 =  VAL (HG$(1)): GOTO 310: REM  GET K
     M AND CHECK
```

Table A.34 cont'd.

```
255  GOTO 252
260  HOME : PRINT "DO YOU WANT TO INPUT ALTITUDE IN KM"
262  PRINT "  OR AS A NUMBER FROM 1 TO 20?"
264  INPUT "INPUT K)M  OR N)UMBER ";Q$
270  IF  LEFT$ (Q$,1) = "K" GOTO 300
280  IF  LEFT$ (Q$,1) = "N" GOTO 405
290  GOTO 260
299  REM  CHECK KM VALUE, FIND HGT NO
300  HOME : INPUT "INPUT ALTITUDE IN KM: ";A1
310  IF A1 > H2 + 30 THEN A1 = H2 + 30: GOTO 360: REM  TOO HIGH
320  FOR I = 1 TO 20
325 A3 = H2 + I * 1.5
330  IF A1 = A3 THEN A2 = I: GOTO 420: REM  GOT HGT NO
340  IF A1 < A3 THEN A1 = A3: GOTO 360: REM  HGT NOT IN FILE
350  NEXT
351  PRINT "ERROR IN HEIGHT LOGIC"
352  STOP
360  REM  DO WE WANT THIS HEIGHT
370  HOME : PRINT "USE ";A1;" KM? ";: INPUT " Y OR N ";Q$
380  IF  LEFT$ (Q$,1) = "N" GOTO 260
390  IF  LEFT$ (Q$,1) = "Y" THEN A2 = (A1 - H2) / 1.5: GOTO 420
400  GOTO 370
405  REM  INPUT HGT NO
410  HOME : INPUT "HGT. NO. FROM 1 TO 20? ";A2
415  IF A2 < 1 OR A2 > 20 GOTO 410
420  REM  CONFIRM BEFORE TRANSFER
425 HG$(PS) =  STR$ (A2 * 1.5 + H2)
430  HOME : PRINT H$(PS); SPC( 4);F$(PS)
432  PRINT T1$(PS);" TO ";T2$(PS);" AT ";HG$(PS);" KM"
435  INPUT "USE THIS DATA? Y OR N ";Q$
440  IF  LEFT$ (Q$,1) = "Y" GOTO 480
445  IF  LEFT$ (Q$,1) = "N" GOTO 455
450  GOTO 430
455  HOME : PRINT "DO YOU WANT TO BACK UP TO INPUT OF"
460  INPUT "D)ATE  F)ILE NAME  T)IMES  A)LTITUDE ";Q$
461  IF  LEFT$ (Q$,1) = "D" GOTO 15
462  IF  LEFT$ (Q$,1) = "F" GOTO 30
463  IF  LEFT$ (Q$,1) = "T" GOTO 130
464  IF  LEFT$ (Q$,1) = "A" GOTO 260
465  GOTO 455
480  REM  TRANSFER INTO ARRAY
490  IF PS <  > 1 GOTO 800
500  GOSUB 5000: REM  FILL ARRAY
505  GOSUB 9450: REM  WINDOW
660  REM  INFO ON FIRST FILE SAVE IN STRINGS
665  REM  READY TO START SECOND PASS
700 PS = 2
710  HOME : INPUT "MORE DATA FROM SAME FILE? Y OR N ";Q$
720  IF  LEFT$ (Q$,1) = "N" THEN  GOSUB 8500: GOTO 11
730  IF  LEFT$ (Q$,1) = "Y" GOTO 742
740  GOTO 710
742  REM  FIX SECOND PASS STRINGS
743 F$(2) = F$(1):H$(2) = H$(1)
750  HOME : INPUT "SAME DATA AND CONDITIONING? Y OR N ";Q$
755  IF  LEFT$ (Q$,1) = "N" GOTO 100
760  IF  LEFT$ (Q$,1) = "Y" GOTO 770
765  GOTO 750
770  REM  SET SECOND PASS STUFF TO FIRST
775 T1$(2) = T1$(1):T2$(2) = T2$(1):HG$(2) = HG$(1):M(2) = M(1)
777 CD$(2) = CD$(1):WN$(2) = WN$(1)
778 SD(2) = SD(1):NV(2) = NV(1)
780  FOR I = 1 TO M(1):G(2,I) = G(1,I): NEXT
785 AU = 1: REM  SET AUTO FLAG
795  GOTO 890: REM  SKIP PASS 2
799  PRINT "ERROR IN PASS HANDLING"
800  REM  THIS IS PASS 2 STUFF
810  IF PS <  > 2 GOTO 799
820  GOSUB 5000: REM  FILL ARRAY
825  GOSUB 9450: REM  WINDOW
```

Table A.34 cont'd.


```
910  REM  OUTPUT ROUTINE HERE
920  HOME : INPUT "OUTPUT TO S)CREEN OR P)RINTER? ";Q$
930  IF  LEFT$ (Q$,1) = "S" GOTO 1500
940  IF  LEFT$ (Q$,1) = "P" GOTO 1000
950  GOTO 920
1000  REM  PRINTER ROUTINE
1050  HOME : PRINT "TURN PRINTER ON"
1090  PRINT D$;"PR#1"
1095  PRINT
1096  PRINT
1097  REM  SET TAB STOP TO 30
1098  PRINT ES$;TB$; CHR$ (31);
1099  PRINT ES$;"1"
1100  PRINT H$(1);TB$;H$(2)
1105  PRINT F$(1);TB$;F$(2)
1107  PRINT HG$(1);" KM";TB$;HG$(2);" KM"
1110  PRINT T1$(1);TB$;T1$(2)
1115  PRINT T2$(1);TB$;T2$(2)
1120  PRINT CD$(1);TB$;CD$(2)
1125  PRINT WN$(1);TB$;WN$(2)
1130  IF NV(1) = 1 THEN  PRINT "NOISY VALUES USED";
1140  IF NV(2) = 1 THEN  PRINT TB$;"NOISY VALUES USED";
1150  IF NV(1) = 1 OR NV(2) = 1 THEN  PRINT
1160  IF NM = 1 THEN  PRINT "NORMALIZED BY ";S8
1169  REM  CLEAR OLD TAB,SET NEW ONE
1170  PRINT TB$;ES$;"2";ES$;TB$; CHR$ (7);ES$;"1";
1171  PRINT ES$;TB$; CHR$ (1);
1172 LM = 22:RM = 72:XS = 47: REM  LEFT,RIGHT,CENTER
1173 SC = 20: REM  SCALE
1175  IF AU = 1 GOTO 1195
1180  FOR I =  - M(1) + 1 TO  - 1
1182 J =  - I
1185 AR = R1(J): GOSUB 9800
1190  NEXT I
1195  FOR I = 0 TO M(2) - 1
1200 AR = R(I): GOSUB 9800
1205  NEXT I
1207  PRINT ES$;"2";: REM  CLEAR TABS
1210  PRINT : PRINT D$;"PR#0"
1215  GOTO 2000
1500  REM  SCREEN OUT
1510  POKE 34,3: POKE 35,20: REM  WINDOW
1512  GOSUB 9770: REM  START PAGING
1515  IF AU = 1 GOTO 1550
1520  FOR I =  - M(1) + 1 TO  - 1
1522  IF LC = 15 THEN  GOSUB 9700: REM  PAGE
1525 J =  - I
1530  PRINT I,R1(J)
1532 LC = LC + 1
1540  NEXT
1550  FOR I = 0 TO M(2) - 1
1552  IF LC = 15 THEN  GOSUB 9700: REM  PAGE
1560  PRINT I,R(I)
1562 LC = LC + 1
1570  NEXT
1575  GOSUB 9700
2000  REM  DONE WITH INPUT
2010  GOSUB 8100: REM  RESTORE WINDOW
2020  HOME : INPUT "USE THE LAST FILE AGAIN? Y OR N ";Q$
2030  IF  LEFT$ (Q$,1) = "N" GOTO 8000
2040  IF  LEFT$ (Q$,1) = "Y" GOTO 2060
2050  GOTO 2020
2060 PS = 1:F$(1) = F$(2):H$(1) = H$(2):AU = 0
2070  GOTO 100
5000  REM  FILL ARRAY ROUTINE
5005 DT$ =  LEFT$ (F$(PS),1): REM  DATA TYPE
5020 M(PS) = T2 - T1 + 1: REM  M(PS)=LENGTH OF PASS PS
5030 I1 = T1 - T0 - 1
5031  IF DT$ <  > "V" THEN NV(PS) = 0: GOTO 5040
```

Table A.34 cont'd.

```
5032  HOME : INPUT "USE NOISY VELOCITY VALUES? Y OR N ";Q$
5033  IF  LEFT$ (Q$,1) = "Y" THEN NV(PS) = 1: GOTO 5040
5034  IF  LEFT$ (Q$,1) = "N" THEN NV(PS) = 0: GOTO 5040
5035  GOTO 5032
5040 S9 = 0:S8 = 0
5041  FOR I = 1 TO M(PS)
5050 G(PS,I) =  VAL (A$(A2,I1 + I))
5060  IF G(PS,I) > 8000 AND NV(PS) = 0 THEN G(PS,I) = 0
5065  IF G(PS,I) > 8000 AND NV(PS) = 1 THEN G(PS,I) = G(PS,I) - 9000
5066 S9 = S9 + G(PS,I)
5067 S8 = S8 + G(PS,I) * 2
5070  NEXT
5075 S8 = S8 / M(PS)
5076 S9 = S9 / M(PS)
5077 SD(PS) =  SQR (S8 - S9 * 2): REM  SAVE STD. DEV.
5080  REM  READY FOR CONDITIONING
5100  REM  CONDITION CHOICES BASED ON DATA TYPE
5115  IF DT$ <  > "C" GOTO 5200
5120  REM  THIS PASS HAS CORRELATION DATA
5125  GOTO 9300: REM  NO CONDITIONING
5200  IF DT$ <  > "V" GOTO 5500
5205  REM  THIS PASS HAS VELOCITY DATA
5210  HOME : INPUT "SUBTRACT LOCAL AVERAGE? Y OR N ";Q$
5215  IF  LEFT$ (Q$,1) = "Y" GOTO 9100
5220  IF  LEFT$ (Q$,1) = "N" GOTO 5230
5225  GOTO 5210
5230  HOME : INPUT "USE ABSOLUTE VALUES? Y OR N ";Q$
5235  IF  LEFT$ (Q$,1) = "Y" GOTO 9200
5240  IF  LEFT$ (Q$,1) = "N" GOTO 5250
5245  GOTO 5230
5250  GOTO 9300: REM  NO CONDITIONING
5500  IF DT$ <  > "P" GOTO 5900
5501  REM  THIS PASS HAS POWER DATA
5502  HOME : INPUT "SUBTRACT LOCAL AVERAGE? Y OR N ";Q$
5503  IF  LEFT$ (Q$,1) = "Y" GOTO 9100
5504  IF  LEFT$ (Q$,1) = "N" GOTO 5510
5505  GOTO 5502
5510  HOME : INPUT "SUBTRACT FILE MINIMUM? Y OR N ";Q$
5515  IF  LEFT$ (Q$,1) = "Y" GOTO 9500
5520  IF  LEFT$ (Q$,1) = "N" GOTO 5530
5525  GOTO 5510
5530  HOME : INPUT "SUBTRACT LOCAL MINIMUM? Y OR N ";Q$
5535  IF  LEFT$ (Q$,1) = "Y" GOTO 9150
5540  IF  LEFT$ (Q$,1) = "N" GOTO 5550
5545  GOTO 5530
5550  GOTO 9300: REM  NO CONDITIONING
5900  PRINT "ERROR IN CONDITIONING LOOP": STOP
6000  REM  CATALOG ROUTINE
6005  REM  SET TEXT WINDOW
6010  POKE 34,10: POKE 35,24
6020  ONERR  GOTO 8400
6025  HOME
6050  PRINT D$;"CATALOG"
6055  GOSUB 8100: REM  RESTORE WINDOW
6060  GOTO 21
7000  REM  CORRELATION ROUTINE
7010  REM  PASS PS DATA OF LENGTH M(PS)
7030  REM  COMPUTE AVERAGE OF G(1,I)*G(2,I+LG)
7050  REM  POSITIVE LAGS IN ARRAY R, NEGATIVE IN R1
7090  POKE 216,0: REM  RESTORE ERROR MESSAGES
7095  HOME : PRINT "COMPUTING POSITIVE LAGS"
7110  FOR LG = 0 TO M(2) - 1
7120 R(LG) = 0
7130  IF M(1) <  = M(2) - LG THEN UL = M(1)
7140  IF M(1) > M(2) - LG THEN UL = M(2) - LG
7150  FOR I = 1 TO UL
7160 R(LG) = R(LG) + G(1,I) * G(2,I + LG)
7170  NEXT
7180 R(LG) = R(LG) / UL
```

Table A.34 cont'd.

```
7190  NEXT
7191  IF AU = 1 OR M(1) = 1 GOTO 7300: REM  SKIP NEG. LAGS FOR AUTO OR NO
      OVERLAP
7195  HOME : PRINT "COMPUTING NEGATIVE LAGS"
7210  FOR NL = 1 TO M(1) - 1
7215 R1(NL) = 0
7220 LL = 1 + NL
7230  IF M(1) < = M(2) + NL THEN UL = M(1)
7240  IF M(1) > M(2) + NL THEN UL = M(2) + NL
7250  FOR I = LL TO UL
7260 R1(NL) = R1(NL) + G(1,I) * G(2,I - NL)
7270  NEXT
7280 R1(NL) = R1(NL) / (UL - NL)
7290  NEXT
7300  RETURN : REM  CORRELATION DONE
8000  REM  QUIT ROUTINE
8001  REM  FIX TEXT WINDOW
8002  POKE 34,0: POKE 35,24
8003  HOME
8005  VTAB (12): PRINT "& AND RETURN TO RESTART"
8010  END
8100  REM  SET TEXT WINDOW TO PRESERVE CATALOG
8105  POKE 34,3: POKE 35,9
8110  RETURN
8200  REM  HEADER MISMATCH ROUITNE
8210 E$ = "CAN'T FIND DISK " + H$(PS)
8220  HOME : PRINT E$
8230  INPUT "T)RY AGAIN OR Q)UIT ";Q$
8240  IF  LEFT$ (Q$,1) = "Q" GOTO 8000
8250  IF  LEFT$ (Q$,1) = "T" GOTO 11
8260  GOTO 8220
8300  REM  ERROR WHILE READING FILE
8310 E$ = "ERROR WHILE READING " + F$(PS)
8320  GOTO 8220
8400  REM  ERROR WHILE CATALOGING
8410  GOSUB 8100: REM  RESTORE WINDOW
8420 E$ = "ERROR TRYING TO CATALOG"
8430  GOTO 8220
8500  REM  FORCE GARBAGE COLLECTION
8505  HOME : PRINT "GARBAGE COLLECTION"
8510  FOR I = 1 TO 20
8515  FOR J = 1 TO 120
8520 A$(I,J) = " "
8525  NEXT
8530  NEXT
8535 I =  FRE (0)
8540  RETURN
8600  REM  S,D,V ROUTINE
8601  IF PS = 1 GOTO 8606
8602  HOME : INPUT "SAME S,D,V PARAMETERS? Y OR N ";Q$
8603  IF  LEFT$ (Q$,1) = "Y" GOTO 8620
8604  IF  LEFT$ (Q$,1) = "N" GOTO 8606
8605  GOTO 8602
8606  HOME : PRINT "WHERE SHOULD I LOOK FOR DATA?"
8610  INPUT "SLOT,DRIVE,VOLUME?  ";SL,DR,VL
8615  IF SL < 4 OR SL > 6 OR DR < 1 OR DR > 2 OR VL < 0 GOTO 8606
8620  RETURN
8650  REM  DATE ROUTINE
8655  IF PS = 1 GOTO 8680
8660  HOME : INPUT "SAME DATE? Y OR N ";Q$
8665  IF  LEFT$ (Q$,1) = "Y" THEN H$(2) = H$(1): GOTO 8685
8670  IF  LEFT$ (Q$,1) = "N" GOTO 8680
8675  GOTO 8660
8680  HOME : INPUT "DATE? (E.G. MARCH 14. 1981) ";H$(PS)
8685  RETURN
9000  REM  SUBTRACT S9 FROM ARRAY
9010  FOR I = 1 TO M(PS)
9020 G(PS,I) = G(PS,I) - S9
9030  NEXT
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.34 cont'd.

```
9040  RETURN
9100  REM  FIND LOCAL AVE
9107 CD$(PS) = "LOCAL AVERAGE SUBTRACTED"
9135  GOTO 9000: REM  SUBTRACT S9
9150  REM  FIND LOCAL MIN
9155 S9 = 9999
9157 CD$(PS) = "LOCAL MINIMUM SUBTRACTED"
9160  FOR I = 1 TO M(PS)
9165  IF G(PS,I) < S9 THEN S9 = G(PS,I)
9170  NEXT
9180  GOTO 9000
9200  REM  ABSOLUTE VALUE
9202 CD$(PS) = "ABSOLUTE VALUE"
9205  FOR I = 1 TO M(PS)
9210 G(PS,I) =  ABS (G(PS,I))
9215  NEXT
9220  RETURN
9300  REM  NO CONDITIONING
9305 CD$(PS) = "NO CONDITIONING"
9310  RETURN
9450  REM  WINDOW ROUTINE
9451  HOME : INPUT "HAMMING WINDOW? Y OR N ";Q$
9452  IF  LEFT$ (Q$,1) = "N" THEN WN$(PS) = "RECTANGULAR": RETURN
9453  IF  LEFT$ (Q$,1) = "Y" GOTO 9455
9454  GOTO 9451
9455 WN$(PS) = "HAMMING"
9460 M1 = M(PS) - 1
9462 M2 = 2 * 3.14159265 / M1
9465  FOR I = 0 TO M1
9470 W = .54 - .46 *  COS (I * M2)
9475 G(PS,I + 1) = G(PS,I + 1) * W
9480  NEXT
9485  RETURN
9500  REM  REMOVE FILE MIN
9505 CD$(PS) = "FILE MINIMUM SUBTRACTED"
9510 S9 = 100 * S2
9515  GOTO 9000
9600  REM  NORMALIZE
9602 NM = 1: REM  SET NORMALIZATION FLAG
9603  HOME : PRINT "NORMALIZATION"
9605 S8 = SD(1) * SD(2)
9610  FOR LG = 0 TO M(2) - 1
9615 R(LG) = R(LG) / S8
9620  NEXT
9625  IF AU = 1 GOTO 9645: REM  SKIP FOR AUTO
9630  FOR NL = 1 TO M(1) - 1
9635 R1(NL) = R1(NL) / S8
9640  NEXT
9645  RETURN : REM  NORMALIZATION DONE
9700  REM  SCREEN OUT PAGING
9702  VTAB (23): PRINT "HIT ANY KEY TO CONTINUE";
9705  IF  PEEK ( - 16384) < 128 GOTO 9705
9710  POKE  - 16368,0
9715  VTAB (23): CALL  - 868
9770 LC = 0
9775  HOME
9778  RETURN
9800  REM  PRINT ONE LINE
9805  PRINT I;TB$;AR;
9807  IF NM = 0 THEN  PRINT : RETURN
9810 HT =  INT (AR * SC + XS + .4999)
9815  IF HT > RM THEN HT = RM:CH$ = "$": GOTO 9830
9820  IF HT < LM THEN HT = LM:CH$ = "$": GOTO 9830
9825 CH$ = "*"
9830  PRINT ES$;TB$; CHR$ (HT + 1);CH$;
9835  IF HT = XS THEN  PRINT
9840  IF HT <  > XS THEN  PRINT ES$;TB$; CHR$ (XS + 1);"."
9845  RETURN
```

this work but is a general purpose tool. The major limitation of the program as shown here is an inability to send output to the HP-9830 for plotting. The Spinwriter can be used to print a low-resolution plot of the correlation, however. Extensive use of screen formatting and messages keep the operator informed of the progress of the lengthy array manipulations and calculations. The compiled version of this program, CSCORR3.OBJ, is used to increase execution speed.

The main loop of the program, from line 11 to line 2070, is primarily a sequence of input requests which control the use of subroutines to access the correct file, select the desired height and time slice from that height, condition the data (e.g., subtract the mean), correlate, and print the results. The main loop is executed twice with the flag PS indicating the number of the pass. The PS flag controls which sections of the main loop are used. For example, on the second pass the operator has the choice of using the height and times from the first pass. This option clearly should not be available on the first pass. After the data from the second pass are conditioned the correlation is performed and the output sent either to the screen or the printer.

The remainder of the progr n consists of routines which implement the functions specified in the main loop. The correlation routine in lines 7000 to 7300 calculates positive and negative lags separately so that the negative lags can be skipped when performing an autocorrelation. The subroutines in lines 9000 to 9645 perform the conditioning which is controlled from operator input in lines 5000 to 5900. The conditioning choices depend on the type of data being processed. They include subtraction of the local minimum, subtraction of the local average, subtraction of the file minimum, or use of the absolute value of the data. After the data are conditioned

they can also be multiplied by a Hamming window in lines 9450 to 9485 to reduce the sidelobes due to truncation of the input sequence. The routines in lines 8000 to 8685 handle various error conditions and procedures which are external to the correlation process. Finally, the routines in lines 9700 to 9800 are used for the output of data to the screen and to the printer. To create the plot on the printer the data is scaled and a character is printed at the corresponding tab position. Values which exceed the limits of the plot are indicated with a different character. Only the numeric output is printed on the screen.

The program SCATTER, shown in Table A.35, generates a table of numbers from which a scatter plot of power variations versus velocity variations is made. The area under the power spectrum in small groups of frequency bins for a matching power and velocity data set are compared. The scaling factors stored in the FFTP and FFTV files are removed before printing the table. The table consists of a small header to indicate the source of the table followed by several lines indicating a range of frequency bins, the area under the power spectrum of the velocity, and the area under the power spectrum of the power. The calculation and printing occur in lines 3000 to 3080. FFTV and FFTP files are identical in form so the routine in lines 7000 to 7055 is used to read both types of files. The parameter FT indicates the file type. The print and read routines are called from the main loop in lines 1000 to 1097. For each time in the time list created in lines 200 to 240 the spectrum of the velocity is read, then the spectrum of the power is read and finally the table is printed. The frequency bin groupings are determined by the DATA statements in lines 98 and 99. The data for a selected frequency bin group for all of the pairs of spectra produce the scatter plot.

Table A.35 SCATTER.

```
1   REM  THIS PROGRAM PRINTS OUT
2   REM  UNNORMALIZED POWER SPECTRA FOR
3   REM  POWER AND VELOCITY DATA SO
4   REM  A SCATTER PLOT CAN BE MADE
5   REM  KEN GIBBS
6   REM  6/29/82
10  REM  ! INTEGER TN,NT,I1,I,NP(1,20)
11  REM  ! INTEGER CN(2,11)
12  REM  ! INTEGER I2
50 D$ =  CHR$ (4)
60  DIM FT(10),F$(2),NP(1,20),FU(1,20)
65  DIM B(1,20,65)
70  DIM CN(2,11)
75  FOR I = 1 TO 11
80  FOR I1 = 1 TO 2
85  READ CN(I1,I)
90  NEXT
95  NEXT
98  DATA  1,3,4,6,7,8,9,11,12,14,15,19
99  DATA  20,22,24,29,30,37,38,43,44,65
100  INPUT "S)TART  Q)UIT ? ";Q$
105  IF  LEFT$ (Q$,1) = "Q" GOTO 9000
110  IF  LEFT$ (Q$,1) <  > "S" GOTO 100
112  ONERR  GOTO 8000
115  PRINT D$;"CATALOG S4"
200  REM  GET TIMES
205 NT = 0
210  INPUT "TIME (E.G. 1029) ? ";TM
215  IF TM = 0 AND NT = 0 GOTO 9000
220  IF TM = 0 AND NT <  > 0 GOTO 240
225 NT = NT + 1
230 FT(NT) = TM
235  IF NT < 10 GOTO 210
240  REM  GOT TIMES
1000  REM  MAIN LOOP
1005 TN = 1
1010 F$(0) = "FFTV/" +  STR$ (FT(TN))
1015 F$(1) = "FFTP/" +  STR$ (FT(TN))
1020  ONERR  GOTO 1090
1025  PRINT D$;"VERIFY ";F$(0)
1030  PRINT D$;"VERIFY ";F$(1)
1035  REM  FOUND BOTH FILES
1040  REM  READ VELOCITY SPECTRUM FIRST
1045 FT = 0
1050  GOSUB 7000
1055  IF  VAL (A$(4)) < 120 GOTO 1090
1056  FOR I = 1 TO 20
1057  IF NP(0,I) >  = 100 GOTO 1060
1058  NEXT
1059  GOTO 1090
1060  REM  GET POWER SPECTRUM
1065 FT = 1
1070  GOSUB 7000
1075  REM  DONE WITH DISK FOR NOW
1080  POKE 216,0
1085  GOSUB 3000: REM  PRINT
1090  IF TN = NT THEN  PRINT D$;"PR#0": RUN
1095 TN = TN + 1
1097  GOTO 1010
3000  REM  PRINT ROUTINE
3005  REM  FIRST A HEADER
3010  PRINT D$;"PR#1"
3015  PRINT : PRINT : PRINT
3020  PRINT A$(1)
3025 M3 =  VAL (A$(3))
3026  PRINT A$(2); RIGHT$ ( STR$ (100 + M3),2)
3030  PRINT
3031  FOR I = 1 TO 20
3032 FU(0,I) = FU(0,I) * 10000
3033 FU(1,I) = FU(1,I) * 1E15
3034  NEXT
3035  FOR I = 1 TO 20
3040  IF NP(0,I) < 100 GOTO 3075
3045  PRINT  VAL (A$(8)) + I * 1.5;" KM"
3050  FOR I1 = 1 TO 11
3051 T0 = 0:T1 = 0
3055  FOR I2 = CN(1,I1) TO CN(2,I1)
3060 T0 = T0 + B(0,I,I2):T1 = T1 + B(1,I,I2)
3065  NEXT
3070  PRINT CN(1,I1);"-";CN(2,I1),T0 / FU(0,I),T1 / FU(1,I)
3072  NEXT
3073  PRINT
3075  NEXT
3080  RETURN
7000  REM  READ IN ONE FFT
7001  PRINT D$;"OPEN ";F$(FT)
7002  PRINT D$;"READ ";F$(FT)
7005  FOR I = 1 TO 8
7010  INPUT A$(I)
7015  NEXT
7020  FOR I = 1 TO 20
7025  INPUT NP(FT,I)
7030  INPUT FU(FT,I)
7035  FOR I1 = 1 TO 65
7040  INPUT B(FT,I,I1)
7045  NEXT
7050  NEXT
7052  PRINT D$;"CLOSE ";F$(FT)
7055  RETURN
8000  REM  CATALOG ERROR
8005  POKE 216,0
8010  PRINT "CAN'T CATALOG S4"
8015  GOTO 100
9000  REM  QUIT
9005  PRINT "& AND RETURN TO RESTART"
9010  END
```

A.14 Generate and Plot Sums of FFTP, FFTV and FFTC Files

The program FFTPVCSUM1.1, shown in Table A.36, produces files which are the summation of FFTP, FFTV or FFTC files. The different types of files are not mixed, however. This program is functionally equivalent to the NUMPOINTS3 program which sums measurable velocity statistics. The program either creates a new sum file in memory or reads a sum file from disk. A selected group of data files are read and added to the sum file. Finally, the sum file is stored on disk. Unlike NUMPOINTS3, FFTPVCSUM1.1 does not differentiate between hours, i.e., the data for every hour within a given month are added together. The heights are summed separately. The file structure is discussed below.

The file structure of the FFTPVCSUM1.1 output files, shown in Table A.37, is similar to the structure of the input FFTP, FFTV and FFTC files. The base height and number of heights correspond to the values for the NUMPOINTS3 files. As discussed above, these values can be modified for non-mesospheric data by changing the assignment statements in the routine which creates new sum files. The header is followed by groups of data which correspond to the first half of a power spectrum for each altitude in the sum file. Each spectrum has associated with it a set of three numbers indicating the sum of the number of valid points from which the spectrum was calculated, the number of possible points, and the number of files contributing to the totals. These numbers are used by the plotting programs to determine if the spectrum is of sufficient quality to plot.

The FFTSUMPLOT3.SRC program and HP-9830 FFTSUMPLOT program, shown respectively in Tables A.38 and A.39, are similar to the programs FFTPVCPLOT.SRC and HP-9830 FFTPVC PLOT shown above for the plotting of the

Table A.36 FFTPVCSUM1.1.


```
1  REM  SUM UP FFTS OF PVC
2  REM  7/9/82
3  REM  MARY OZARKA
10  REM  !INTEGER I,J,IH,TN,TM,NT,PF,FP
11  REM  ! INTEGER E(20,65),NA(20),FT(10)
20 D$ =  CHR$ (4)
30  DIM SA(25,65),N(25,3),E(20,65),LS$(45)
31  DIM NA(20),FF(20)
35  REM  GET MONTH,YEAR
40  INPUT "MONTH? ";M$
50  INPUT "YEAR? ";YR
60  IF YR < 1978 OR YR > 1989 GOTO 50
70  INPUT "SUM FILE S,D,V ?  ";DS,DD,DV
80  IF DS < 4 OR DS > 6 OR DD < 1 OR DD > 2 OR DV < 0 GOTO 70
84  ONERR  GOTO 70
85  PRINT D$;"CATALOG S";DS;",D";DD;",V";DV
90  INPUT "TYPE OF DATA? (P,V,C)  ";TP$
100 TY$ =  LEFT$ (TP$,1)
110  REM  NF$ IS NAME OF SUM FILE
120 NF$ = "FFT" + TY$ + "/" + M$ + "." +  STR$ (YR)
130  ONERR  GOTO 1100
140  PRINT D$;"VERIFY ";NF$
145  POKE 216,0
150  PRINT D$;"OPEN ";NF$
160  PRINT D$;"READ ";NF$
170  INPUT M1$
180  INPUT Y1
190  INPUT TS$
200  INPUT HD
210  INPUT NH
240  REM  READ IN THE ARRAY
260  FOR I = 1 TO NH
265  INPUT N(I,1)
270  INPUT N(I,2)
272  INPUT N(I,3)
275  FOR J = 1 TO 65
280  INPUT SA(I,J)
290  NEXT
300  NEXT
310  PRINT D$;"CLOSE ";NF$
345  REM  INITIALIZE COUNTER PF BEFORE GETTING DATA FILE
346 PF = 0
350  INPUT "DATA FILE S,D,V ?  ";SS,SD,SV
355  IF SS < 4 OR SS > 6 OR SD < 1 OR SD > 2 OR SV < 0 GOTO 350
360  ONERR  GOTO 350
370  PRINT D$;"CATALOG S";SS;",D";SD;",V";SV
375  REM  GET TIMES
380 NT = 0
390  INPUT "TIME (E.G. 1029) ?  ";TM
400  IF TM = 0 AND NT = 0 GOTO 960
410  IF TM = 0 AND NT <  > 0 GOTO 500
420 NT = NT + 1
430 FT(NT) = TM
440  IF NT < 10 GOTO 390
500 TN = 0
510 TN = TN + 1
520  REM  FILE NAME FOR INPUT IS F$
530 F$ = "FFT" + TY$ + "/" +  STR$ (FT(TN))
535  ONERR  GOTO 750
540  PRINT D$;"VERIFY ";F$
545  POKE 216,0
550  PRINT D$;"OPEN ";F$
560  PRINT D$;"READ ";F$
570  INPUT L$: INPUT WC: INPUT WC
575  REM  N3 IS THE NUMBER OF POINTS IN THE SOURCE
580  INPUT N3
590  INPUT WC: INPUT WC: INPUT WC
595  REM  HS IS THE BASE HEIGHT FOR THE SOURCE
600  INPUT HS
```

Table A.36 cont'd.

```
610  REM  INPUT STATEMENTS TO READ ARRAY
620  FOR IH = 1 TO 20
623  INPUT NA(IH)
625  INPUT FF(IH)
630  FOR I = 1 TO 65
640  INPUT E(IH,I)
650  NEXT
660  NEXT
670  PRINT D$;"CLOSE ";F$
671  IF  MID$ (M$,3,1) < > MID$ (L$,3,1) OR  MID$ (M$,2,1) < > MID$ (
     L$,2,1) THEN  PRINT "CAN'T USE THIS DATA": GOTO 760
672 Y2 =  VAL ( MID$ (L$, LEN (L$) - 8,4))
673  IF Y2 <  > YR THEN  PRINT "CAN'T USE THIS DATA": GOTO 760
674 UF = 0
675  REM  ADD INPUT FILE TO SUM FILE
680 HN = (HS - HD) / 1.5
685  FOR IH = 1 TO 20
690 HN = HN + 1
695  IF HN > NH GOTO 745
700  IF HN < 1 GOTO 740
705 N(HN,1) = N(HN,1) + NA(IH)
710 N(HN,2) = N(HN,2) + N3
712 N(HN,3) = N(HN,3) + 1
715  FOR I = 1 TO 65
720 SA(HN,I) = SA(HN,I) + E(IH,I)
730  NEXT I
735 UF = 1
740  NEXT IH
745  REM  DONE WITH INPUT FILE
746  IF UF = 1 THEN PF = PF + 1:LS$(PF) = L$ + " " +  STR$ (FT(TN))
750  IF TN < NT GOTO 510
760  INPUT "DO YOU WANT TO CONTINUE ? (Y/N)  ";L2$
770  IF L2$ <  > "Y" THEN  GOTO 810
780  INPUT "SAME S,D,V FOR INPUT ? (Y/N)  ";L3$
790  IF L3$ = "Y" THEN  GOTO 360
800  GOTO 350
810  ONERR  GOTO 970
820  PRINT D$;"OPEN ";NF$;",S";DS;",D";DD;",V";DV
830  PRINT D$;"DELETE ";NF$
840  PRINT D$;"OPEN ";NF$
850  PRINT D$;"WRITE ";NF$
851  PRINT M1$
852  PRINT Y1
853  PRINT TS$
854  PRINT HD
855  PRINT NH
860  FOR I = 1 TO NH
865  PRINT N(I,1)
870  PRINT N(I,2)
872  PRINT N(I,3)
875  FOR J = 1 TO 65
880  PRINT SA(I,J)
890  NEXT
900  NEXT
910  PRINT D$;"CLOSE ";NF$
930  FOR I = 1 TO PF
940  PRINT LS$(I)
950  NEXT
960  END
970  PRINT "CANNOT WRITE TO DISK"
975  POKE 216,0
980  INPUT "WANT TO TRY AGAIN? (Y/N)  ";L4$
990  IF  LEFT$ (L4$,1) = "Y" GOTO 810
995  END
1100  PRINT "CANNOT READ SUMFILE"
1105  POKE 216,0
1110  PRINT "C)REATE SUM FILE, T)RY AGAIN, R)E-ENTER MONTH, YEAR, S,D,V O
     R Q)UIT"
1120  INPUT "C, T, R, OR Q ?  ";Q$
```

Table A.36 cont'd.

```
1130  IF  LEFT$ (Q$,1) = "C" GOTO 1160
1135  IF  LEFT$ (Q$,1) = "T" GOTO 130
1140  IF  LEFT$ (Q$,1) = "R" GOTO 40
1145  IF  LEFT$ (Q$,1) = "Q" GOTO 960
1148  GOTO 1120
1160  REM  CREATE NEW SUMFILE IN MEMORY
1210 M1$ = M$
1220 Y1 = YR
1230 TS$ = "FFT" + TY$ + "SUM"
1240 HD = 57
1250 NH = 25
1260  FOR I = 1 TO NH
1265 N(I,1) = 0
1270 N(I,2) = 0
1272 N(I,3) = 0
1275  FOR J = 1 TO 65
1280 SA(I,J) = 0
1285  NEXT
1290  NEXT
1310  GOTO 350
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.37 Apple II disk format for FFTPVCSUM files.

```
Sequential text file, Apple DOS 3.3
File name e.g. FFTV/MAY.1978

M1$:   month
Y1:    year
TS$:   "FFTPSUM" for the sum of FFTP files, for example
HD:    57 = base altitude, first altitude at 58.5 km., etc.
NH:    25 = number of heights

N(1,1):    total no. of valid data points used to calculate the
             spectra in the files summed at the lowest altitude
N(1,2):    total no. of possible points for the lowest altitude
N(1,3):    no. of files summed at the lowest altitude
SA(1,1):     DC component of summed spectrum at lowest altitude
  ...........
SA(1,65):    component of the spectrum at the folding frequency for
               the lowest altitude

N(2,1):    valid data points at the 2nd altitude
N(2,2):    possible data points at the 2nd altitude
N(2,3):    no. of files at the 2nd altitude
SA(2,1):     DC component of summed spectrum at the 2nd altitude
  ...........
SA(2,65):    folding frequency component at the 2nd altitude

           ....................
           ....................

N(NH,1):   valid data points at the highest altitude
N(NH,2):   possible data points at the highest altitude
N(NH,3):   no. of files at the highest altitude
SA(NH,1):    DC component of summed spectrum at the highest altitude
  ...........
SA(NH,65):   folding frequency component at the highest altitude
```

Table A.38 FFTSUMPLOT3.SRC.


```
1  REM  READS THE FFTS FILES
2  REM  AND TRANSFERS THEM TO HP FOR PLOT
3  REM  RUDY REAVIS
4  REM  07:20:82:16:30
15 D$ =  CHR$ (4): REM  CTRL-D FOR DOS
20  DIM N(3),SA(65),S$(50),M1$(15),TS$(50)
30 VD = 5 / 6
50  REM  GET FILE NAME & OPEN FILE
60  PRINT : PRINT : PRINT "C)ATALOG F)ILE INPUT Q)UIT"
65 K = 0
70  INPUT "C,F OR Q?";Q$
80  IF  LEFT$ (Q$,1) = "Q" GOTO 1100
90  IF  LEFT$ (Q$,1) = "F" GOTO 120
100  IF  LEFT$ (Q$,1) = "C" GOTO 1200
110  GOTO 60
120  INPUT "FILE NAME?";F$
130  ONERR  GOTO 1000
140  PRINT D$,"VERIFY";F$: POKE 216,0
150  PRINT D$;"OPEN";F$: PRINT D$;"READ";F$
160  REM
161  REM  GET HEADER DATA
170  INPUT M1$: REM  MONTH AS A STRING
172  INPUT Y1: REM  YEAR AS A REAL
174  INPUT TS$: REM  TITLE STRING
176  INPUT BH: REM  BASE HEIGHT
178  INPUT NH: REM  NO. OF HEIGHTS
179  PRINT D$;"CLOSE";F$
180  GOSUB 2500
192  GOSUB 2000: REM  SET-UP VIA (6522)
195  GOSUB 500: REM  SEND HEADER TO HP
200  FOR HT = 1 TO NH
210  INPUT N(1): INPUT N(2): INPUT N(3)
215  FOR I1 = 1 TO 65: INPUT SA(I1): NEXT
220  IF N(1) = 0 GOTO 315
225  IF N(1) / N(2) < VD GOTO 315
226  IF HT < K THEN 315
230  REM
231  REM  SEND THIS DATA @ HEIGHT HT
232  REM
239  REM  TELL HP THAT DATA IS COMING
240 S$ = "1": GOSUB 800
244  REM  SEND HEIGHT
245 S$ =  STR$ (BH + 1.5 * HT): GOSUB 800
250  FOR I2 = 1 TO 3
253 S$ =  STR$ (N(I2))
255  GOSUB 800
280  NEXT
285  FOR I1 = 1 TO 65
286 S$ =  STR$ (SA(I1) / N(3))
290  GOSUB 800
300  NEXT
315  NEXT
316 S$ = "0": GOSUB 800: REM  HALT HP
317  PRINT D$;"CLOSE";F$
320  REM  DONE-RESTART
325  RUN
330  STOP
500  REM  SEND HEADER ROUTINE
517  PRINT : PRINT "START HP NOW"
530 S$ = M1$: GOSUB 800
535 S$ =  STR$ (Y1): GOSUB 800
540 S$ = TS$: GOSUB 800
555  RETURN
797  REM
798  REM  SEND S$ TO HP SUBROUTINE
799  REM
800 A1 =  LEN (S$)
805  FOR A2 = 1 TO A1
810  IF  PEEK ( - 15475) < > 18 THEN 810
```

Table A.38 cont'd.

```
820  POKE  - 15488, ASC ( MID$ (S$,A2,1)): NEXT
830  IF  PEEK ( - 15475) <  > 18 THEN 830
840  POKE  - 15488,10
850  REM  -15488 IS B DATA REGISTER
860  RETURN
1000  PRINT "NO SUCH FILE ON DISK": GOTO 60
1100  PRINT : PRINT "& AND RETURN TO RESTART": END
1200  ONERR  GOTO 1220
1210  PRINT D$;"CATALOG": POKE 216,0: GOTO 60
1220  PRINT "ERROR TRYING TO CATALOG": POKE 216,0: GOTO 60
2000  REM  INIT VIA PORT B OF 6522
2005  POKE  - 15486,255: REM   DDRB=$FF
2010  POKE  - 15476,136: REM   PCR=$88
2015  POKE  - 15475,255: REM   CLEAR IFR
2020  POKE  - 15474,127: REM   CLEAR IER
2050  RETURN
2500  PRINT "BASE=";BH
2510  INPUT "DO ALL HEIGHTS (Y OR N)? ";H$
2520  IF H$ = "Y" THEN 2550
2525  IF H$ <  > "N" THEN 2510
2530  PRINT "ALT=BASE + (1.5 * K)  K=1,2,...,20"
2535  INPUT "START AT HEIGHT (INPUT K)? ";K
2540  IF K < 0 OR K > 20 THEN 2530
2550  PRINT D$;"OPEN";F$
2555  PRINT D$;" READ ";F$
2560  INPUT M1$
2565  INPUT Y1
2570  INPUT TS$
2575  INPUT BH
2580  INPUT NH
2999  RETURN
```

Table A.39 HP-9830 FFTSUMPLOT.

```
10 REM PROGRAM TO PLOT FFTSUM FILES
20 REM KEN GIBBS 7/21/82
21 REM WITH THANKS TO RUDY
30 REM PROGRAM SHOULD BE RUN WHEN INDICATED BY APPLE
40 DIM N[3],S[65]
50 DIM M$[25],T$[10],Y$[4]
55 DIM H$[10],L$[40]
60 DIM P[32]
100 REM READ HEADER
110 ENTER (2,*)M$
115 REM MONTH AS A STRING
120 ENTER (2,*)Y$
125 REM YEAR AS A STRING
130 ENTER (2,*)T$
135 REM TITLE AS A STRING
200 REM MAKE DATE
210 M$[LEN(M$)+1]=" "
220 M$[LEN(M$)+1]=Y$
230 L$="PLOT OF DATA FILE"
300 REM DEFINE PLOT CONTROL
301 REM P(1),P(2),P(3),P(4) ARE SCALE CONTROL
302 REM PAPER IS 8.5 INCH VERTICAL, 11 INCH WIDE
303 REM P(21)=PAPER HEIGHT/WIDTH
304 REM P(5),P(6),P(7)=BOTTOM,TOP,NO. OF TICS ON VERT. AXIS
305 REM P(8)=HORIZ. POSITION OF VERT. AXIS
306 REM P(9)=SPACING BETWEEN HORIZ. TICS
307 REM P(24),P(18)=LEFT, WIDTH OF HORIZ. AXIS
308 REM P(10),P(11)=H,V COORD. FOR DATE
309 REM P(12),P(13)=H,V FOR ALTITUDE
310 REM P(14),P(15)=H,V FOR PERCENT
311 REM P(16),P(17)=H,V FOR TITLE
312 REM P(19),P(20)=HEIGHT,ASPECT RATIO OF LABELS
313 REM P(22),P(23)=HORIZ. HALF LENGTH,SPACING OF VERT. TICS
314 REM P(25)=HORIZ. SPACE BETWEEN DATA POINTS
315 REM P(26)=VERT.SCALE FOR DATA
316 REM P(27)=PLOT CONTROL 1=DOT,0=LINES
317 REM P(28)=RELATIVE VERT. OF HORIZ. AXIS LABEL TO AXIS
318 REM P(30),P(31)=H,V FOR NO. OF FILES
319 REM P(29)=SIZE OF AXIS LABEL
320 REM P(32)=V FOR FREQ. AXIS
345 REM READ TABLE, ZERO INTO P(I) WHICH ARE CALCULATED
350 FOR I=1 TO 32
355 READ P[I]
360 NEXT I
370 DATA 0,1100,35,850,200,750,10,0,0,75,775,75,700,75,625
372 DATA 50,40,700,2,1.7,0,10,0,300,0,0,0,-25,1,60
374 DATA 550,150
400 REM CALCULATE OTHER PLOT PARAMETERS
404 REM DUMMY TIC FOR NOW
405 P[9]=4*P[18]/(65-1)
409 REM PUT VERT AXIS AT LEFT OF HORIZ. AXIS
410 P[8]=P[24]
414 REM FIND VERT TIC SPACING
415 P[23]=(P[6]-P[5])/P[7]
419 REM PAPER RATIO USE SCALE STUFF
420 P[21]=(P[4]-P[3])/(P[2]-P[1])
424 REM CALCULATE HORIZ SPACING FOR DATA
425 P[25]=P[18]/(65-1)
429 REM CALCUALTE VERT. MULTIPLIER FOR DATA
430 P[26]=(P[6]-P[5])/999
450 REM FIRST CHARACTER SENT IS (J)UMP
451 REM J1=0 MEANS NO MORE DATA
452 REM J1=1 GET READY FOR DATA
455 ENTER (2,*)J1
457 IF J1=0 THEN 1500
460 REM READ HEIGHT
465 ENTER (2,*)H$
470 REM READ DATA CONSTANTS
472 FOR I=1 TO 3
474 ENTER (2,*)N[I]
478 NEXT I
480 REM READ DATA POINTS
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.39 cont'd.

```
482 FOR I=1 TO 65
484 ENTER (2,*)S[I]
486 NEXT I
490 REM FIX HEIGHT STRING
492 H$[LEN(H$)+1]=" KM"
500 REM START PLOT
505 SCALE P[1],P[2],P[3],P[4]
510 YAXIS P[5],P[9],P[24],P[24]+P[18]
524 REM TIC VERT AXIS
525 FOR I=1 TO P[7]
527 XAXIS P[5]+I*P[23],+1,P[8],P[8]-P[22]
528 XAXIS P[5]+I*P[23],-1,P[8],P[8]+P[22]
529 NEXT I
580 PLOT P[10],P[11],1
581 LABEL (*,P[19],P[20],0,P[21])M$
585 PLOT P[12],P[13],1
586 LABEL (*)H$
587 FORMAT F4.0," %"
590 PLOT P[14],P[15],1
591 LABEL (587)100*N[1]/N[2]
592 PLOT P[30],P[31],1
593 LABEL (594)N[3]
594 FORMAT F3.0," FILES"
595 PLOT P[16],P[17],1
596 LABEL (*)L$[1,8];T$;L$[8,17]
600 REM PLOT DATA
610 FOR I=2 TO 65
655 REM FIND HORIZONTAL POSITION
660 X=P[24]+(I-1)*P[25]
665 REM FIND VERTICAL POSITION
670 Y=S[I]*P[26]+P[5]
675 REM PUT PEN DOWN AFTER MOVING
680 PLOT X,Y,-2
685 REM CHECK PLOT CONTROL FOR DOTS
690 IF P[27]=0 THEN 700
695 PEN
700 NEXT I
702 PEN
705 GOTO 1130
1000 REM LABEL HORIZ. AXIS
1001 LABEL (*,P[29],P[20],0,P[21])
1005 Y=P[5]+P[28]
1006 PLOT P[24]+P[18],Y,1
1007 CPLOT 1.5,0
1008 LABEL (*)"M/S"
1010 FOR I=2 TO -2 STEP -1
1015 I1=5*I
1020 PLOT P[24]+I1*P[9],Y,1
1025 CPLOT P[I+3],0
1030 LABEL (1035)I1
1035 FORMAT F3.0
1040 NEXT I
1060 REM FREQ. AXIS
1065 XAXIS P[32],P[18]/4,P[24]-P[18],P[24]+P[18]
1066 Y=P[32]+P[28]
1067 PLOT P[24]+P[18],Y,1
1068 CPLOT 1.5,0
1069 LABEL (*)"HZ"
1070 FOR I=4 TO -4 STEP -1
1075 PLOT P[24]+I*P[18]/4,Y,1
1080 IF I >= 0 THEN 1095
1085 CPLOT -1,0
1090 GOTO 1100
1095 CPLOT -1.3,0
1100 LABEL (1105)I
1105 FORMAT F2.0
1110 NEXT I
1130 REM READY FOR MORE?
1135 REM SHOULD LOOP BACK HERE AFTER OK
1140 DISP "PREPARE PLOTTER--TYPE CONT/EXEC"
1145 STOP
1150 GOTO 450
1500 END
```

FFTP, FFTV, and FFTC files. The FFTSUM file is read from disk and the heights of interest are sent to the HP-9830. The horizontal axis labels for the plotting routine have not been implemented in the programs shown here.

A.15 Batch Processing of DGEN3 and PVCFFTGEN5.1 Programs

The program MAKE EXEC 1.7, shown in Table A.40, generates an execute file called NIGHT to control batch processing of the DGEN3 and PVCFFTGEN5.1 programs shown above. One set of volumes on the Corvus hard disk is reserved for the minute-by-minute data disks and a second set is reserved for the processed data. The NIGHT file causes the Apple II to repeatedly run the DGEN3 and PVCFFTGEN5.1 programs using each data volume as input and a corresponding volume for output. The operation is entirely automatic and therefore can be done at night when the Apple II is available for a long period of time. The complete processing of 6 days of minute-by-minute data requires about 5 to 6 hours. The operator must transfer the data from floppy disks to the Corvus disk before starting and the results must be copied to a second set of floppy disks when done. This disadvantage is outweighed by the time savings of unattended operation at night.

MAKE EXEC 1.7 allows up to 10 volumes to be processed at one time. In general the number of disks which can be processed is limited by the number of volumes available on the Corvus disk. The data disks are copied onto a set of volumes with the date and time of the files noted for later entry into the MAKE EXEC 1.7 program. These volumes do not have to be contiguous. A special floppy disk is copied onto an equivalent number of volumes which will receive the output data. This special floppy disk should contain only a dummy date file. When this special disk is copied to a Corvus volume it will clear the volume and leave a date file which can be changed to match the date file on one of the data volumes. Recall that the coherent-scatter

Table A.40 MAKE EXEC 1.7.


```
10  REM  MAKE EXEC
12 D$ =  CHR$ (4)
14  PRINT D$;"OPEN NIGHT"
16  PRINT D$;"DELETE NIGHT"
30  REM  !INTEGER L,D,K,F
40  DIM A$(200)
50 D = 0
60  INPUT "NUMBER OF DAYS ? (UP TO 10)  ";L
63  IF L = 0 THEN  GOTO 220
64  PRINT " "
65 R$ =  STR$ (D + 1)
66  PRINT "DATA FOR RUN " + R$
70  INPUT "RUN DGEN3.OBJ ? Y OR N  ";A$(0 + D)
80  INPUT "RUN PVCFFTGEN5.OBJ ? Y OR N  ";A$(10 + D)
90  IF A$(0 + D) = "Y" THEN  GOTO 100
95  IF A$(10 + D) <  > "Y" THEN  GOTO 220
100  INPUT "DATE (E.G. APRIL 14. 1978) ?  ";A$(20 + D)
110  INPUT "SOURCE ? (S,D,V)  ";A$(30 + D),A$(40 + D),A$(50 + D)
120  INPUT "DESTINATION ? (S,D,V)  ";A$(60 + D),A$(70 + D),A$(80 + D)
130 K = 0
140  INPUT "START TIME ? (E.G. 1029)  ";A$(90 + D + K * 10)
150 K = K + 1
160  IF A$(90 + D + (K - 1) * 10) <  > "0" THEN  GOTO 140
180 D = D + 1
190  IF L <  > D THEN  GOTO 64
210  GOTO 260
220  PRINT "PROGRAM EXITED"
230 E$ =  STR$ (D)
240  PRINT E$ + " DAY(S) OF DATA WILL BE PROCESSED
250  IF E$ = "0" GOTO 410
260  PRINT D$;"OPEN NIGHT"
261  PRINT D$;"WRITE NIGHT"
265  PRINT "BLOAD RUNTIME,S6,V9"
270 RF = 0
275  FOR D = 0 TO L - 1
280  IF A$(0 + D) <  > "Y" GOTO 305
285 RF = RF + 1
290  IF RF = 1 THEN  PRINT "BRUN DGEN3.OBJ,S6,V9"
295  IF RF > 1 THEN  PRINT "&"
300  GOSUB 420
305  NEXT
310 RF = 0
315  FOR D = 0 TO L - 1
320  IF A$(10 + D) <  > "Y" GOTO 350
325 RF = RF + 1
330  IF RF = 1 THEN  PRINT "BRUN PVCFFTGEN5.1.OBJ,S6,V9"
335  IF RF > 1 THEN  PRINT "&"
340  GOSUB 420
345  PRINT "Y"
350  NEXT
410  PRINT D$;"CLOSE"
415  END
420  PRINT A$(20 + D)
430  PRINT A$(30 + D);",";A$(40 + D);",";A$(50 + D)
440  PRINT A$(60 + D);",";A$(70 + D);",";A$(80 + D)
450 K = 0
460  PRINT A$(90 + D + K * 10)
470 K = K + 1
480  IF A$(90 + D + (K - 1) * 10) <  > "0" THEN  GOTO 460
490  PRINT "Y"
500  PRINT "Y"
510  RETURN
```

data is organized on a one date per disk basis. The date and Corvus volume number should be noted for entry into the MAKE EXEC 1.7 program. It is important to verify that each data volume has a matching output volume and that the output volume has the correct date file and no other files. Any errors which occur while the NIGHT file is in control of the Apple II will effectively end useful operation.

The operator must indicate to MAKE EXEC 1.7 the number of volumes to be processed and for each volume whether programs DGEN3 and PVCFFTGEN5.1 are to be used to process the data. The date, source and destination locations, and the file times are then entered. The input of times is similar to that for the NUMPOINTS3 program shown above. Any time which does not correspond to a file will be ignored at runtime. A time entry of 0 terminates the time input loop. After collecting the input information MAKE EXEC 1.7 generates the NIGHT file on the most recently accessed disk volume. The NIGHT file is a text file which is not easily transferred from volume to volume under Apple DOS. Difficulties were encountered while trying to execute the NIGHT file from the Corvus disk. Therefore, MAKE EXEC 1.7 is loaded from the floppy disk. The command EXEC NIGHT with the appropriate slot, drive and volume parameters causes the NIGHT file to begin controlling the Apple II. Note that the NIGHT file expects the compiler RUNTIME library and the compiled version of the programs DGEN3 and PVCFFTGEN5.1 to be on volume 9 of the Corvus disk. If this is not the case then lines 265, 290 and 330 must be modified.

A.16 Height-to-Height Correlation

The program CS HGT CORR, shown in Table A.41, is a specialized version of the CSCORRSRC3 program shown above in Table A.34. CS HGT CORR calculates and prints a table of the cross-correlation of minute-by-minute velocity

Table A.41 CS HGT CORR.


```
1  REM  CS HEIGHT CORRELATOR-KEN GIBBS 5/5/82
4  REM  ! INTEGER I,J,N3,HH,HL,HT,H6,H7,N5,PS,AU
5  DIM F$(2),H$(2),A(20,120),HV(20),HA(20)
8 D$ =  CHR$ (4):PS = 1:AU = 0
10  HOME : PRINT "CS HEIGHT CORRELATOR  5/5/82": GOSUB 8100
11  GOSUB 8600: REM  S,D,V
15  GOSUB 8650: REM  DATE
16  ONERR  GOTO 8200
17  PRINT D$;"VERIFY ";H$(PS);",S"; STR$ (SL);",D"; STR$ (DR);",V"; STR$
     (VL)
20  REM  HAVE CORRECT FLOPPY
21  HOME : INPUT "C)ATALOG  F)ILE INPUT  Q)UIT  ";Q$
22  IF  LEFT$ (Q$,1) = "Q" GOTO 8000
23  IF  LEFT$ (Q$,1) = "F" GOTO 30
24  IF  LEFT$ (Q$,1) = "C" GOTO 6000
25  GOTO 21
30  HOME : INPUT "INPUT FILE NAME  ";F$(PS)
32  HOME : PRINT "READING FILE ";F$(PS)
34  ONERR  GOTO 8300
35  PRINT D$;"VERIFY ";F$(PS)
45  PRINT D$;"OPEN ";F$(PS): PRINT D$;"READ ";F$(PS)
46  REM  GET THE HEADER STUFF
47  INPUT L$: INPUT H[illegible]: INPUT M3: INPUT N3
49  INPUT S2: INPUT L2: INPUT A6
50  INPUT H2
51  REM  GET DATA ARRAY
52  IF H2 >  = 87 OR H2 <  = 31.5 THEN  PRINT "NO DATA OF INTEREST": PRINT
     D$;"CLOSE": GOTO 8230
54  REM  GET DATA ARRAY
55  FOR I = 1 TO 20
60  FOR J = 1 TO 120
65  INPUT A(I,J)
66  IF A(I,J) > 8000 THEN A(I,J) = 0
70  NEXT
75  NEXT
80  PRINT D$;"CLOSE ";F$(PS)
90  HOME : PRINT "DONE WITH INPUT"
100  REM  FIND HEIGHT LIMITS
160  IF H2 >  = 60 GOTO 180
165 HL = (60 - H2) / 1.5
170  IF HL / 2 =  INT (HL / 2) THEN HH = 20: GOTO 195
175  IF HL / 2 <  >  INT (HL / 2) THEN HH = 19: GOTO 195
180 HH = 20 - (H2 - 60) / 1.5
185  IF HH / 2 =  INT (HH / 2) THEN HL = 2: GOTO 195
190  IF HH / 2 <  >  INT (HH / 2) THEN HL = 1
195  REM  GOT HGT LIMITS
200  REM  FIND AVERAGES AND SUBTRACT
202  HOME : PRINT "FINDING MEANS AND VARIANCES"
205  FOR HT = HL TO HH STEP 2
210 AV = 0:VR = 0:NP = 0
212  REM  CALCULATE MEAN AND VARIANCE
215  FOR I = 1 TO N3
220  IF A(HT,I) = 0 GOTO 245
225 T1 = A(HT,I)
230 AV = AV + T1
235 VR = VR + T1 * T1
240 NP = NP + 1
245  NEXT
247  IF NP = 0 THEN NP = 1
250 AV = AV / NP
255 HV(HT) = VR / NP - AV * AV
260 HA(HT) = AV
265  REM  REMOVE MEAN
275  FOR I = 1 TO N3
280  IF A(HT,I) = 0 GOTO 295
285 A(HT,I) = A(HT,I) - AV
290  IF A(HT,I) = 0 THEN A(HT,I) = 1
295  NEXT
300  NEXT
```

ORIGINAL PAGE IS
OF POOR QUALITY

Table A.41 cont'd.

```
400  REM  PRINTER INIT
410  HOME : PRINT "TURN PRINTER ON"
420  PRINT D$;"PR#1"
430  PRINT H$(PS)
435  PRINT F$(PS)
440  PRINT N3;"  MINUTES OF DATA"
445  PRINT
450  PRINT "Z","DZ","NORM","N","RHO"
500  REM  OUTER LOOP IS BASE HEIGHT
510  FOR H6 = HL TO HH - 2 STEP 2
515  PRINT
520  REM  INNER LOOP DETERMINES HEIGHT OFFSET
530  FOR H7 = H6 + 2 TO HH STEP 2
540  GOSUB 7000: REM  CORRELATION
550  GOSUB 1000: REM  PRINT A LINE
560  NEXT
570  NEXT
580  REM  DONE WITH FILE
590  PRINT : PRINT
595  PRINT D$;"PR#0"
600 AU = 1
605  GOTO 11
1000  REM  PRINT A LINE
1005 Z = H2 + H6 * 1.5
1010 DZ = (H7 - H6) * 1.5
1015 NF =  SQR (HV(H6) * HV(H7))
1017  IF NF = 0 THEN RH = 0: GOTO 1025
1020 RH = X / NF
1022 NF = NF / 10000
1025  PRINT Z,DZ,NF,N5,RH
1096  RETURN
6000  REM  CATALOG ROUTINE
6005  REM  SET TEXT WINDOW
6010  POKE 34,10: POKE 35,24
6020  ONERR  GOTO 8400
6025  HOME
6050  PRINT D$;"CATALOG"
6055  GOSUB 8100: REM  RESTORE WINDOW
6060  GOTO 21
7000  REM  CORRELATION ROUTINE
7030  REM  COMPUTE AVERAGE OF A(Z,I)*A(Z+Z1,I)
7040  REM  IF EITHER VALUE IS 0 THEN SKIP
7050  REM  SAVE THE NUMBER OF PRODUCTS
7090  POKE 216,0: REM  RESTORE ERROR MESSAGES
7100  REM  INIT
7105 N5 = 0
7110 X = 0
7200  REM  LOOP OVER MINUTES
7205  FOR I = 1 TO N3
7210  IF A(H6,I) = 0 OR A(H7,I) = 0 GOTO 7225
7215 X = X + A(H6,I) * A(H7,I)
7220 N5 = N5 + 1
7225  NEXT
7227  IF N5 = 0 GOTO 7240
7230 X = X / N5
7240  RETURN
8000  REM  QUIT ROUTINE
8001  REM  FIX TEXT WINDOW
8002  POKE 34,0: POKE 35,24
8003  HOME
8005  VTAB (12): PRINT "& AND RETURN TO RESTART"
8010  END
8100  REM  SET TEXT WINDOW TO PRESERVE CATALOG
8105  POKE 34,3: POKE 35,9
8110  RETURN
8200  REM  HEADER MISMATCH ROUITNE
8210 E$ = "CAN'T FIND DISK " + H$(PS)
8220  HOME : PRINT E$
8230  INPUT "T)RY AGAIN OR Q)UIT ";Q$
```

Table A.41 cont'd.

```
8240  IF  LEFT$ (Q$,1) = "Q" GOTO 8000
8250  IF  LEFT$ (Q$,1) = "T" GOTO 11
8260  GOTO 8220
8300  REM  ERROR WHILE READING FILE
8310 E$ = "ERROR WHILE READING " + F$(PS)
8320  GOTO 8220
8400  REM  ERROR WHILE CATALOGING
8410  GOSUB 8100: REM  RESTORE WINDOW
8420 E$ = "ERROR TRYING TO CATALOG"
8430  GOTO 8220
8600  REM  S,D,V ROUTINE
8601  IF AU = 0 GOTO 8606
8602  HOME : INPUT "SAME S,D,V PARAMETERS? Y OR N ";Q$
8603  IF  LEFT$ (Q$,1) = "Y" GOTO 8620
8604  IF  LEFT$ (Q$,1) = "N" GOTO 8606
8605  GOTO 8602
8606  HOME : PRINT "WHERE SHOULD I LOOK FOR DATA?"
8610  INPUT "SLOT,DRIVE,VOLUME?  ";SL,DR,VL
8615  IF SL < 4 OR SL > 6 OR DR < 1 OR DR > 2 OR VL < 0 GOTO 8606
8620  RETURN
8650  REM  DATE ROUTINE
8655  IF AU = 0 GOTO 8680
8660  HOME : INPUT "SAME DATE? Y OR N ";Q$
8665  IF  LEFT$ (Q$,1) = "Y" GOTO 8685
8670  IF  LEFT$ (Q$,1) = "N" GOTO 8680
8675  GOTO 8660
8680  HOME : INPUT "DATE? (E.G. MARCH 14. 1981) ";H$(PS)
8685  RETURN
```

data at various heights within a single file. Specifically, the data at heights which are an integral multiple of 3 km are correlated with those heights above it at 3 km intervals. The 3 km interval corresponds to the height resolution of the Urbana coherent-scatter radar. The results are plotted by hand.

The operation of CS HGT CORR is straightforward because it only uses a single file from disk at one time and performs the same operations each time. The inputs required are therefore limited to slot, drive and volume of the data source (typically a floppy disk), and the file name to process. A catalog of the data disk which can be printed on the screen is saved by using the screen text-window functions. The program loops back after processing a file to accept another file name to process. The main loop consists of lines 11 to 605.

Once the file name is entered the program will process the file without further input. The file is read from disk in lines 45 to 80. Note that the velocity values calculated in noise which have been offset by the PDP-15 are set to zero. The heights which satisfy the requirements given above are detemined in lines 100 to 195. The mean and variance for these heights is determined in lines 200 to 260 and subtracted in lines 265 to 300. Finally, a file identifier is printed on the Spinwriter followed by a table heading. The program loops over allowable heights and height differences, printing a line containing height, delta height, the normalization factor, the number of non-zero products in the correlation, and the correlation. The normalization factor is the square root of the product of the variances.

The remainder of the program consists of routines to handle disk errors, input date and disk parameters, handle the screen window, and perform the cross-correlation. After the first file has been processed the

operator is given the choice of using the same slot, drive and volume parameters and the same date. The AU variable set to 1 indicates that a file has already been processed. The program could be improved by forcing the screen display of the catalog to be updated whenever a new date or disk is requested.

REFERENCES

Atlas, D., J. I. Metcalf, J. H. Richter and E. E. Gossard [1970], The birth of "CAT" and microscale turbulence, J. Atmos. Sci., 27, 903-913.

Backof, C. A., Jr. and S. A. Bowhill [1974], Collection and processing of data from a phase-coherent meteor radar, Aeron. Rep. No. 59, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana-Champaign.

Batchelor, G. K. [1953], The Theory of Homogeneous Turbulence, Cambridge University Press, Cambridge.

Beer, T. [1974], Atmospheric Waves, Halsted Press, New York.

Booker, J. R. and F. P. Bretherton [1967], The critical layer for internal gravity waves in a shear flow, J. Fluid Mech., 27, 513-539.

Booker, H. G. and W. E. Gordon [1950], A theory of radio scattering in the troposphere, Proc. IRE, 38, 401-412.

Breeding, R. J. [1972], A nonlinear model of the break-up of internal gravity waves due to their exponential growth with height, J. Geophys. Res., 77, 2681-2692.

Bretherton, F. P. [1966], The propagation of groups of internal gravity waves in shear flow, Quart. J. Roy. Meteorol. Soc., 92, 466-480.

Carter, D. A., B. B. Balsley and W. L. Ecklund [1980], The Poker Flat MST radar: an outline of the primary echoing mechanisms and a description of the facility, in Proceedings of the 19th Conference on Radar Meteorology, 563-567, American Meteorological Society, Boston.

Chandrasekar, S. [1961], Hydrodynamic and Hydromagnetic Instability, Oxford University Press, New York.

Corcos, G. M. and F. S. Sherman [1976], Vorticity concentration and the dynamics of unstable free shear layers, J. Fluid Mech., 73, 241-264.

Countryman, I. D. and S. A. Bowhill [1979], Wind and wave observations in the mesosphere using coherent-scatter radar, Aeron. Rep. No. 89, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana-Champaign.

Davis, P. A. and W. R. Peltier [1979], Some characteristics of the Kelvin-Helmholtz and resonant overreflection modes of shear flow instability and of their interaction through vortex pairing, J. Atmos. Sci., 36, 2394-2412.

Drazin, P. G. [1958], The stability of a shear layer in an unbounded heterogeneous inviscid fluid, J. Fluid Mech., 4, 211-224.

Einaudi, F. and D. P. Lalas [1973], On the growth rate of an unstable disturbance in a gravitationally stratified shear flow, J. Atmos. Sci., 30, 1707-1710.

Evans, J. V. [1969], Theory and practice of ionosphere study by Thomson scatter radar, Proc. IEEE, 57, 496-530.

Fritts, D. C. [1978], The nonlinear gravity wave-critical level interaction, J. Atmos. Sci., 35, 397-413.

Fritts, D. C. [1979], The excitation of radiating waves and Kelvin-Helmholtz instabilities by the gravity wave-critical level interaction, J. Atmos. Sci., 36, 12-23.

Fritts, D. C. and M. A. Geller [1976], Viscous stabilization of gravity wave critical level flows, J. Atmos. Sci., 33, 2276-2284.

Fukao, S., T. Sato, S. Kato, R. M. Harper, R. F. Woodman and W. E. Gordon [1979], Meospheric winds and waves over Jicamarca on 23-24 May 1974, J. Geophys. Res., 84, 4379-4386.

Fukao, S., T. Sato, R. Harper and S. Kato [1980], Radio wave scattering from the tropical mesosphere observed with the Jicamarca radar, Radio Sci., 15, 447-457.

Gage, K. S. and B. B. Balsley [1980], On the scattering and reflection mechanisms contributing to clear air radar echoes from the troposphere, stratosphere, and mesosphere, Radio Sci., 15, 243-257.

Gage, K. S., B. B. Balsley and J. L. Green [1981], Fresnel scattering model for the specular echoes observed by VHF radar, Radio Sci., 16, 1447-1453.

Gage, K. S. and J. L. Green [1978], Evidence for specular reflection from monostatic VHF radar observations of the stratosphere, Radio Sci., 13, 991-1001.

Geller, M. A., H. Tanaka and D. C. Fritts [1975], Production of turbulence in the vicinity of critical levels for internal gravity waves, J. Atmos. Sci., 32, 2125-2135.

Gibbs, K. P. and S. A. Bowhill [1979], The Urbana coherent-scatter radar: systhesis and first results, Aeron. Rep. No. 90, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana-Champaign.

Gossard, E. E. and J. H. Richter [1970], Internal waves in the atmosphere from high resolution radar measurements, J. Geophys. Res., 75, 3523-3536.

Gossard, E. E. and K. C. Yeh [1980], Foreword, Radio Sci., 15, 147-150.

Gregory, J. B. [1961], Radio wave reflections from the mesosphere. I. Heights of occurrence. J. Geophys. Res., 66, 429-445.

Herrington, L. J. and S. A. Bowhill [1983], Phase modulating the Urbana radar, Aeron. Rep. No. 109, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana, Champaign.

Hines, C. O. and C. A. Reddy [1967], On the propagation of atmospheric gravity waves through regions of wind shear, J. Geophys. Res., 72, 1015-1034.

Gage, K. S. and B. B. Balsley [1980], On the scattering and reflection mechanisms contributing to clear air radar echoes from the troposphere, stratosphere, and mesosphere, Radio Sci., 15, 243-257.

Gage, K. S., B. B. Balsley and J. L. Green [1981], Fresnel scattering model for the specular echoes observed by VHF radar, Radio Sci., 16, 1447-1453.

Gage, K. S. and J. L. Green [1978], Evidence for specular reflection from monostatic VHF radar observations of the stratosphere, Radio Sci., 13, 991-1001.

Geller, M. A., H. Tanaka and D. C. Fritts [1975], Production of turbulence in the vicinity of critical levels for internal gravity waves, J. Atmos. Sci., 32, 2125-2135.

Gibbs, K. P. and S. A. Bowhill [1979], The Urbana coherent-scatter radar: systhesis and first results, Aeron. Rep. No. 90, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana-Champaign.

Gossard, E. E. and J. H. Richter [1970], Internal waves in the atmosphere from high resolution radar measurements, J. Geophys. Res., 75, 3523-3536.

Gossard, E. E. and K. C. Yeh [1980], Foreword, Radio Sci., 15, 147-150.

Gregory, J. B. [1961], Radio wave reflections from the mesosphere. I. Heights of occurrence, J. Geophys. Res., 66, 429-445.

Harrington, T. A. and M. A. Geller [1975], Performance of the University of Illinois meteor-radar system - A preliminary report, Aeron. Rep. No. 65, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana-Champaign.

Herrington, L. J. and S. A. Bowhill [1983], Phase modulating the Urbana radar, Aeron. Rep. No. 109, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana, Champaign.

Hines, C. O. and C. A. Reddy [1967], On the propagation of atmospheric gravity waves through regions of wind shear, J. Geophys. Res., 72, 1015-1034.

Hinze, J. O. [1959], Turbulence, McGraw-Hill Co., New York.

Hill, R. J. and S. A. Bowhill [1976], Small-scale fluctuations in D-region ionization due to hydrodynamic turbulence, Aeron. Rep. No. 75, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana-Champaign.

Hodges, R. R., Jr. [1967], Generation of turbulence in the upper atmosphere by internal gravity waves, J. Geophys. Res., 72, 3455-3458.

Horton, B. E. and S. A. Bowhill [1971], Computer simulation of supersonic rarefied gas flow in the transition region, about a spherical probe; a Monte Carlo approach with application to rocket-borne ion probe experiments, Aeron. Rep. No. 43, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana-Champaign.

Howard, L. N. [1961], Note on a paper of John W. Miles, J. Fluid Mech., 10, 509-512.

Jones, W. L. [1968], Reflection and stability of waves in stably stratified fluids with shear flow: a numerical study, J. Fluid Mech., 34, 609-624.

Jones, W. L. and D. D. Houghton [1972], The self-destructing internal gravity wave, J. Atmos. Sci., 29, 844-849.

Jordan, E. C. and K. G. Balmain [1968], Electromagnetic Waves and Radiating Systems, Prentice-Hall, Inc., Englewood Cliffs, New Jersey.

Klostermeyer, J. [1980], Computation of acoustic-gravity waves, Kelvin-Helmholtz instabilities, and wave-induced eddy transport in realistic atmospheric models, J. Geophys. Res., 85, 2829-2839.

Klostermeyer, J. and R. Ruster [1980], Radar observation and model computation of a jet stream-generated Kelvin-Helmholtz instability, J. Geophys. Res., 85, 2841-2846.

Lalas, D. P. and F. Einaudi [1976], On the characteristics of gravity waves generated by atmospheric shear layers, J. Atmos. Sci., 33, 1248-1259.

Lindzen, R. S. and A. J. Rosenthal [1976], On the instability of Helmholtz velocity profiles in stably stratified fluids when a lower boundary is present, J. Geophys. Res., 81, 1561-1571.

Miles, J. W. [1961], On the stability of heterogeneous shear flows, J. Fluid Mech., 10, 496-508.

Miller, K. L., S. A. Bowhill, K. P. Gibbs and I. D. Countryman [1978], First measurements of mesospheric vertical velocities by VHF radar at temperate latitudes, Geophys. Res. Lett., 5, 939-942.

Oppenheim, A. V. and R. W. Schafer [1975], Digital Signal Processing, Prentice Hall, Englewood Cliffs, New Jersey.

Ottersten, H. [1969], Radar backscattering from the turbulent clear atmosphere, Radio Sci., 4, 1251-1255.

Patnaik, P. C., F. S. Sherman and G. M. Corcos [1976], A numerical simulation of Kelvin-Helmholtz waves of finite amplitude, J. Fluid Mech., 73, 215-240.

Peltier, W. R., J. Halle and T. L. Clark [1978], The evolution of finite amplitude Kelvin-Helmholtz billows, Geophys. Astrophys. Fluid Dynamics, 10, 53-87.

Rastogi, P. K. and S. A. Bowhill [1976a], Scattering of radio waves from the mesosphere - I. Theory and observations, J. Atmos. Terr. Phys., 38, 399-411.

Rastogi, P. K. and S. A. Bowhill [1976b], Scattering of radio waves from the mesosphere - II. Evidence for intermittent mesospheric turbulence, J. Atmos. Terr. Phys., 38, 449-462.

Rastogi, P. K. and J. Rottger [1982], VHF radar observations of coherent reflections in the vicinity of the tropopause, J. Atmos. Terr. Phys., 44, 461-469.

Roth, P. R. [1982] (Ed.), Research in Aeronomy, October 1, 1981 - March 31, 1982, Prog. Rep. 82-1, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana-Champaign.

Rottger, J. [1980], Reflection and scattering of VHF radar signals from atmospheric refractivity structures, Radio Sci., 15, 259-276.

Rottger, J. and C. H. Liu [1978], Partial reflection and scattering of VHF radar signals from the clear atmosphere, Geophys Res. Lett., 5, 357-360.

Rottger, J., P. K. Rastogi and R. F. Woodman [1979], High-resolution VHF radar observations of turbulence structures in the mesosphere, Geophys. Res. Lett., 6, 617-620.

Royrvik, O., K. P. Gibbs and S. A. Bowhill [1982], VHF power scattered from the mesosphere at mid-latitudes, J. Geophys. Res., 87, 2501-2508.

Schmidt, G., R. Ruster and P. Czechowsky [1979], Complementary code and digital filtering for detection of weak VHF signals from the middle atmosphere, IEEE Trans. Geosci. Electr., GE-17, 154-161.

Scotti, R. S. and G. M. Corcos [1972], An experiment on the stability of small disturbances in a stratified free shear layer, J. Fluid Mech., 52, 499-528.

Skolnik, M. I. [1962], Introduction to Radar Systems, McGraw-Hill Book Company, New York.

Tatarskii, V. I. [1971], The Effects of the Turbulent Atmosphere on Wave Propagation, National Tech. Inform. Service, Springfield, Va., 74-76.

Thorpe, S. A. [1973], Experiment of instability and turbulence in a stratified shear flow, J. Fluid Mech., 61, 731-751.

Weiland, R. M. and S. A. Bowhill [1981], Investigation of the winds and electron concentration variability in the D-region of the ionosphere by the partial reflection radar technique, Aeron. Rep. No. 100, Aeron. Lab., Dept. Elec. Eng., Univ. Ill., Urbana-Champaign.

Woodman, R. F. and A. Guillen [1974], Radar observations of winds and turbulence in the stratosphere and mesosphere, J. Atmos. Sci., 31, 493-505.

Woods, J. D. [1969], On Richardson's number as a criterion for laminar-turbulent-laminar transition in the ocean and atmosphere, Radio Sci., 4, 1289-1298.